

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

COMME EXIGENCE PARTIELLE
À L'OBTENTION DE LA
MAÎTRISE EN GÉNIE DE LA PRODUCTION AUTOMATISÉE
M. Ing.

PAR
M. Donavan PRIEUR

RÉALISATION DE CLASSIFICATEURS NEURONIQUES POUR LA
RECONNAISSANCE À BASSE PUISSANCE DE VISAGES DANS DES SÉQUENCES
VIDÉO

MONTREAL, LE 11 JUIN 2012

©Tous droits réservés, Prieur Donavan, 2012

©Tous droits réservés

Cette licence signifie qu'il est interdit de reproduire, d'enregistrer ou de diffuser en tout ou en partie, le présent document. Le lecteur qui désire imprimer ou conserver sur un autre média une partie importante de ce document doit obligatoirement en demander l'autorisation à l'auteur.

PRÉSENTATION DU JURY

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE :

Monsieur Éric Granger, directeur de mémoire
Génie de la production automatisée à l'École de technologie supérieure

Monsieur Claude Thibeault, codirecteur de mémoire
Génie électrique à l'École de technologie supérieure

Monsieur Jean-Marc Robert, président du jury
Génie logiciel et des technologies de l'information à l'École de technologie supérieure

Monsieur Yvon Savaria, membre du jury
Génie électrique à l'École Polytechnique de Montréal

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 8 MAI 2012

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

Je tiens à remercier MM. Éric Granger, Claude Thibeault et Yvon Savaria pour leur patience et leurs judicieux conseils qui m'ont permis de compléter ce projet de recherche.

Je voudrais aussi remercier le Fonds de recherche Nature et technologie, le Conseil de recherche en sciences naturelles et en génie du Canada ainsi que la compagnie Octasic Inc. pour le soutien financier qu'ils ont apporté au projet.

Finalement, je veux remercier mes parents et Vanessa qui ont cru en moi et m'ont soutenu tout au long de mes études.

RÉALISATION DE CLASSIFICATEURS NEURONIQUES POUR LA RECONNAISSANCE À BASSE PUISSANCE DE VISAGES DANS DES SÉQUENCES VIDÉO

Donavan PRIEUR

RÉSUMÉ

La biométrie a été proposée comme solution à la multiplication des codes et mots de passe nécessaire pour sécuriser du matériel électronique mobile. Dans cet ouvrage, une technique pour la réalisation d'un système de reconnaissance de visages dans des séquences vidéo à faible consommation d'énergie est proposée et sa performance dans trois implémentations matérielles est évaluée. Cette technique est basée sur l'entraînement de classificateurs neuroniques de type fuzzy ARTMAP à l'aide d'une stratégie d'apprentissage optimisant simultanément les hyperparamètres, la valeur des poids synaptiques et la structure des réseaux de neurones avec un algorithme d'optimisation multiobjective par essais de particules.

Trois processeurs commercialement offerts ont été retenus pour évaluer la performance de l'implémentation matérielle du système de reconnaissance. Ce sont le Core i3-530 d'Intel, l'Atom N270 lui aussi d'Intel et le Vocallo MGW de la compagnie Octasic. La qualité des prédictions d'identité, l'espace mémoire utilisé, le temps de traitement ainsi que la consommation d'énergie ont été évalués à l'aide de la base de données vidéo IIT-CNRC.

Les résultats ont montré que le temps passé par les nœuds de calcul du processeur Vocallo MGW à communiquer entre eux est négligeable comparativement au temps de traitement total. Il est possible d'estimer la qualité des prédictions du réseau de neurones fuzzy ARTMAP entraîné avec la stratégie d'apprentissage OMOEP en fonction du nombre de caractéristiques retenues dans les motifs d'entrée à l'aide d'un classificateur de type « plus proche voisin ». La croissance exponentielle de la taille des réseaux de neurones en fonction du nombre de caractéristiques présentes dans les motifs d'entrée favorise l'utilisation du plus petit nombre de caractéristiques possible. Finalement, l'utilisation de l'algorithme fuzzy ARTMAP entraîné avec une stratégie OMOEP permet de trouver le meilleur compromis entre la qualité de prédiction et la quantité de ressources utilisées. Cette solution couplée avec un processeur parallèle de basse puissance permet d'obtenir une faible consommation d'énergie tout en maintenant un taux de classification intéressant et un temps de traitement proche du temps réel.

Mots-clés : Reconnaissance, visage, vidéo, parallèle, basse puissance, fuzzy ARTMAP, essaim de particules, multiobjectif, biométrie

RÉALISATION DE CLASSIFICATEURS NEURONIQUES POUR LA RECONNAISSANCE À BASSE PUISSANCE DE VISAGES DANS DES SÉQUENCES VIDÉO

Donavan PRIEUR

ABSTRACT

Biometrics have been suggested as a solution to the multiplication of codes and passwords necessary to secure mobile electronic devices. In this document, a technique for the realisation of a low-power video-based facial recognition system is proposed and the performance of three hardware implementations is evaluated. For an efficient implementation, the training of fuzzy ARTMAP neural classifiers with a training strategy to simultaneously optimize the hyper-parameters, the synaptic weight values and the neural configuration of the networks through multi-objective particle swarm optimization.

Three commercially available processors have been selected to measure the performance of hardware implementation of the system. They are the Core i3-530 from Intel, the Atom N270 also from Intel and the Vocallo MGW from Octasic. The quality of the identity predictions, the memory usage, the processing time and the energy consumption has been evaluated using the IIT-CNRC video database.

Results have shown that the time spent communicating between processing cores is negligible compared to the time spent calculating the identity predictions when using the Vocallo MGW. It is possible to estimate the classification rate of a fuzzy ARTMAP neural classifier trained with the MOPSO training technique for a given numbers of characteristics kept in the input patterns by using a nearest neighbor type classifier. The exponential growth of the network's memory usage when more characteristics are kept suggests using the least amount of characteristics possible. Finally, the use of the fuzzy ARTMAP neural classifier trained with the MOPSO strategy allows finding the best trade-off between the quality of the predictions made and the amount of resources used. This solution paired to a multi-core low power processor provides low energy consumption while maintaining a good classification rate and a processing time close to real-time.

Keywords : Recognition, face, video, parallel, low-power, fuzzy ARTMAP, particle swarm, multi-objective, biometry

TABLE DES MATIÈRES

	Page
INTRODUCTION	1
CHAPITRE 1 IDENTIFICATION BIOMÉTRIQUE	7
1.1 Validation, identification et dépistage	8
1.2 Caractéristiques biométriques ainsi que leurs performances	10
1.3 Système typique de reconnaissance de visages dans des séquences vidéo.....	13
1.4 État de l’art en reconnaissance de visages dans des séquences vidéo	15
1.5 Défis à la réalisation.....	18
1.6 Conclusion	20
CHAPITRE 2 PRÉDICTION DE L’IDENTITÉ.....	23
2.1 Extraction et sélection de caractéristiques avec l’analyse par composantes principales	23
2.2 Classificateur neuronique fuzzy ARTMAP	27
2.2.1 Architecture.....	28
2.2.2 Hyperparamètres	29
2.2.3 Algorithme	30
2.2.4 Analyse de complexité.....	32
2.3 Stratégies d’apprentissage pour les réseaux de neurones fuzzy ARTMAP	33
2.3.1 Optimisation par essaim de particules (OEP)	34
2.3.2 Optimisation multiobjective par essaim de particules (OMOEP).....	36
2.4 Conclusion	40
CHAPITRE 3 IMPLÉMENTATION MATÉRIELLE	43
3.1 Sélection des processeurs.....	43
3.1.1 Intel Core™ i3-530	47
3.1.2 Intel Atom™ N270.....	48
3.1.3 Octasic Vocallo MGW	49
3.2 Modifications algorithmiques	51
3.2.1 Fonction de choix.....	51
3.2.2 L’inversion de la modification du signal de suivis	51
3.2.3 Encodage binaire de la couche F_{AB}	52
3.2.4 Traitement parallèle	52
3.2.5 Prédiction par accumulation d’informations.....	54
3.3 Conclusion	55
CHAPITRE 4 MÉTHODOLOGIE D’ÉVALUATION.....	57
4.1 Base de données expérimentale	57
4.2 Méthodologie expérimentale	59
4.3 Indicateurs de performance.....	60
4.4 Conclusion	61

CHAPITRE 5 RÉSULTATS EXPÉRIMENTAUX	63
5.1 Communications internœuds	63
5.2 Extraction et sélection de caractéristiques	65
5.3 La performance en reconnaissance d'images individuelles.....	68
5.4 La performance en reconnaissance dans des séquences vidéo	74
5.5 Conclusion	76
CONCLUSION	79
ANNEXE I EXEMPLE DE CALCUL DE LA CONSOMMATION D'ÉNERGIE	83
LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES.....	85

LISTE DES TABLEAUX

	Page
Tableau 3.1	Caractéristiques des processeurs commercialement disponibles46
Tableau 5.1	Taux de classification moyen en généralisation sur la base de test71
Tableau 5.2	Espace mémoire moyen occupé par le réseau entraîné.....72
Tableau 5.3	Temps d'exécution moyen pour le traitement de la base de test73
Tableau 5.4	Consommation d'énergie moyenne pour le traitement de la base de test.74

LISTE DES FIGURES

	Page
Figure 1.1	Système typique de reconnaissance de visage dans des séquences vidéo .14
Figure 2.1	Réseau de neurones artificiels fuzzy ARTMAP29
Figure 2.2	Optimisation par essaim de particules (OEP)34
Figure 2.3	Un front de Pareto37
Figure 3.1	La fréquence d’horloge et la consommation d’énergie par instruction pour les processeurs sélectionnés.....44
Figure 3.2	La configuration de la mémoire pour le processeur Intel Core i3-53047
Figure 3.3	La configuration de la mémoire pour le processeur Intel Atom N27049
Figure 3.4	La configuration de la mémoire pour le processeur Vocallo MGW50
Figure 3.5	Les stratégies de communication54
Figure 4.1	Validation croisée modifiée58
Figure 5.1	Temps de traitement et communication pour les différentes stratégies de communication64
Figure 5.2	Le taux de classification en fonction du différent nombre de caractéristiques conservées66
Figure 5.3	Le nombre de neurones dans le réseau entraîné ainsi que le nombre maximum de neurones que peut contenir la mémoire cache du Vocallo MGW en fonction du nombre de caractéristiques conservées.....67
Figure 5.4	Fronts Pareto pour les différentes répliques statistiques en fonction du nombre de caractéristiques conservées dans le motif d’entrée69
Figure 5.5	Fronts Pareto pour les différentes répliques statistiques en fonction du nombre de caractéristiques conservées dans le motif d’entrée (espace mémoire occupé en kB)70
Figure 5.6	Taux de classification moyen en fonction de la taille du tampon de décision75

LISTE DES ALGORITHMES

	Page
Algorithme 2.1 Analyse par composantes principales	26
Algorithme 2.2 Séquence d'apprentissage du réseau de neurones fuzzy ARTMAP	32
Algorithme 2.3 La stratégie d'apprentissage OEP pour fuzzy ARTMAP	35
Algorithme 2.4 La stratégie d'apprentissage OMOEP pour fuzzy ARTMAP	39

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

UTA	Unité de traitement arithmétique
CISA	Circuit intégré spécifique à l'application
ADN	Acide désoxyribonucléique
PSN	Processeur de signal numérique
AGE	Appariement de graphes élastiques
RPRU	Réseau prédiffusé programmable par l'utilisateur
MCM	Modèles cachés de Markov
ACI	Analyse des composantes indépendantes
k-PPV	« k » plus proches voisins
ADL	Analyse du discriminant linéaire
OMOEP	Optimisation multiobjective par essaim de particules
ACP	Analyse des composantes principales
RNBDP	Réseaux de neurones basés sur la densité de probabilité
OEP	Optimisation par essaim de particules
RNBR	Réseaux de neurones à bases radiales
RI	Région d'intérêt
CAA	Cartes autoassociatives
ITGE	Intégration à très grande échelle

LISTE DES SYMBOLES ET UNITÉS DE MESURE

a	Motif d'entrée classificateur
A	Motif d'entrée complétement
b	Motif d'entrée module de suivi
B	Matrice de transformation
c	Constantes de pondération de variation de position d'une particule
d	Nombre de caractéristiques dans un motif de la base de données
E	Matrice de vecteurs propres issue de la matrice de covariance « Ω »
f	Nombre de neurones dans le réseau entraîné
F(s_qⁱ)	Valeur de la fonction objective pour la particule « i » lors de l'itération « q »
F₁	Couche de neurones dans le réseau fuzzy ARTMAP
g	Nombre de solutions conservées dans l'archive
γ_i	Seuil d'indécision pour la classe « i »
g_{max}	Nombre maximum de solutions conservées dans l'archive
J	Indice du neurone gagnant
k	Nombre de caractéristiques conservées dans les motifs d'entrée transformés
L	Nombres d'individus dans la base de données
n	Nombre de motifs d'entrée dans la base de données
O	Ordre de complexité de l'algorithme
P	Nombre de nœuds de calculs
p_g	Position de la meilleure performance globale
p_i	Position de la meilleure performance locale de la particule « i »
q	Numéro de l'itération en cours
q_{max}	Nombre maximum d'itérations pour l'optimisation
r	Valeurs aléatoires
s_qⁱ	Position de la particule « i » lors de l'itération « q »
t	Nombre de motifs dans la base de données de test
T_j(A)	Valeur d'activation du neurone « j » pour le motif d'entrée « A »
U	Nombre de particules utilisées dans l'optimisation par essaims de particules
v_i	Vitesse de la particule « i »
W₁	Matrice de connexions synaptique dans le réseau fuzzy ARTMAP
X	Base de données
Y	Motif d'apprentissage
α	Hyperparamètre de signal
β	Hyperparamètre d'apprentissage
ε	Hyperparamètre de suivi
η	Compression
κ	Consommation énergétique du processeur
λ	Vecteur de valeurs propres issu de la matrice de covariance « Ω »
μ	Vecteur moyen de la base de données « X »
ρ	Hyperparamètre de vigilance
υ	Fréquence d'horloge du processeur
Φ_i(a)	Probabilité que le motif « a » appartient à la classe « i »
χ	Nombre d'opérations exécutées par le processeur
Ω	Matrice de covariance de la base de données « X »

INTRODUCTION

Lorsqu'ils ont introduit le transistor (Bardeen et Brattain, 1948), Bardeen, Brattain et Shockley n'avaient probablement pas conscience de l'impact que cette invention aurait sur la société moderne. Ce nouveau composant électronique a vite supplanté les tubes à vide en raison de leurs petites dimensions, leur bonne efficacité énergétique ainsi que leur faible coût de fabrication. Grâce à ses attributs intéressants, l'utilisation en grand nombre des transistors a généré les deux grandes tendances de l'électronique moderne, c'est-à-dire la miniaturisation et le traitement numérique de l'information.

La première de ces deux tendances, la miniaturisation, est vite devenue apparente. Peu de temps après l'introduction des transistors, le monde de la radio a été bouleversé par l'arrivée de très petites radios alimentées par des piles et qui peuvent être transportées facilement par l'utilisateur, peu importe où il désire aller. Le temps où la famille se réunissait autour de la radio pour l'écouter ensemble a disparu et a fait place à une consommation plus individualiste du média radiophonique. Ce même processus s'est produit dans le monde de la télévision lorsque les transistors ont été couplés aux cristaux liquides permettant ainsi de miniaturiser les moniteurs au point de permettre l'utilisation de petits écrans vidéo portables et alimentés à l'aide de piles dignes des fantasmes technologiques de Dick Tracy.

La deuxième tendance, le traitement numérique de l'information, a mis plus de temps à se propager. Par contre, elle a possiblement transformé la société moderne d'une façon encore plus significative. La capacité des transistors à être regroupés de façon à intégrer de multiples circuits dans un seul composant discret (Kilby, 1976) a rendu possible la production de processeurs capables d'effectuer une série d'opérations mathématiques sur des données. Le faible coût de fabrication de ces processeurs électroniques comparativement à leurs homologues électromécaniques a permis à ce type de systèmes de se répandre et d'être utilisé pour des applications de plus en plus diverses. L'interconnexion des ordinateurs en vastes réseaux informatiques a encore plus accéléré la tendance. De nos jours, presque toute l'information est disponible en format numérique et est accessible en tout temps et peu

importe l'endroit, lorsque l'utilisateur a un accès informatique au réseau global. Ce sont donc ces deux tendances qui sont derrière l'explosion des produits informatiques portables de types miniportatifs, téléphones intelligents ou tablettes multimédias et de la consommation beaucoup plus personnelle de l'information que ces produits ont apportés.

Que ce soit pour des fins de sécurité ou bien de personnalisation de l'expérience de l'utilisateur, ces nouveaux produits informatiques mobiles nécessitent la capacité de différencier les utilisateurs entre eux. Ceci est traditionnellement effectué en validant l'identité soumise par l'utilisateur (nom ou numéro d'utilisateur) à l'aide d'une pièce d'information secrète connue par l'utilisateur (mot de passe) (Jain, Ross et Pankanti, 2006). Puisque la majorité de ces produits informatiques mobiles ont maintenant une caméra vidéo intégrée, il est possible de remplacer ce processus par l'identification automatique des visages dans une séquence vidéo sans avoir à modifier l'appareil en question.

La biométrie consiste à utiliser une caractéristique, qu'elle soit physiologique ou comportementale, afin d'identifier l'utilisateur (Jain, Ross et Prabhakar, 2004). La reconnaissance de visages dans des séquences vidéo offre plusieurs avantages sur la validation d'identité à l'aide d'un mot de passe. En utilisant l'individu lui-même pour les besoins d'identification au lieu de quelque chose qu'il possède ou qu'il connaît, le processus d'identification est fondamentalement moins sujet à la perte, la subtilisation ou la corruption de l'identifiant (Jain, Ross et Pankanti, 2006). Puisque le processus d'identification par reconnaissance de visages dans des séquences vidéo ne nécessite aucune action de la part de l'individu à identifier (Jain, Ross et Prabhakar, 2004), il est possible de « rafraichir » l'identité (Niinuma, Unsang et Jain, 2010) d'une façon tout à fait transparente à l'utilisateur pour bloquer l'usage de l'appareil par un autre individu après l'identification initiale. Avec le nombre grandissant d'enfants utilisant des produits technologiques mobiles, la reconnaissance de visages dans des séquences permet aussi d'étendre le contrôle sur l'utilisation de l'appareil à des individus incapables d'utiliser un mot de passe alphanumérique traditionnel.

Par contre, la tâche de reconnaître un visage dans une séquence vidéo est complexe et requiert un grand nombre de calculs avant de produire la prédiction de l'identité. Cette puissance de calcul généralement fournie par un processeur spécialisé ou bien un processeur générique avec une cadence d'horloge élevée contraste fortement avec les ressources computationnelles limitées que l'on trouve dans les appareils électroniques mobiles. Ces derniers étant alimentés à l'aide d'une batterie rechargeable, la consommation énergétique du processus devient un critère de conception important au même titre que la précision de la prédiction et le temps nécessaire pour la produire. L'utilisation de processeurs avec de multiples nœuds de calcul permet d'augmenter la puissance de calculs tout en limitant la croissance de la consommation d'énergie du processeur (Canturk et al., 2006). Mais le traitement parallèle nécessaire (Constantinou et al., 2005) pour tirer la performance maximale de ce type de processeurs n'est pas présent dans la majorité des algorithmes de reconnaissance de visages dans des séquences vidéo.

Les systèmes d'identification distribués, utilisant des caméras numériques intelligentes branchées sur le réseau informatique pour des fins de contrôle d'accès, présentent des contraintes sur les ressources énergétiques et computationnelles assez similaires à celles des appareils électroniques mobiles pour pouvoir utiliser le même type de solutions.

Les systèmes de reconnaissance de visages dans des séquences vidéo, comme tous les systèmes biométriques adaptatifs (Jain, Ross et Prabhakar, 2004), comportent deux grandes phases d'opération : la phase d'apprentissage et la phase opérationnelle. Lors de l'apprentissage, le système crée un modèle facial, une représentation statistique de l'apparence du visage d'une personne, pour chacun des individus inscrits. Dans la phase opérationnelle, le système utilise ces modèles faciaux pour les comparer à l'image de l'individu à identifier et produire la prédiction de l'identité de ce dernier.

Cette comparaison entre l'image de l'individu et les modèles faciaux est l'opération la plus complexe de la phase opérationnelle. Il est donc important d'utiliser un algorithme efficace pour accomplir la comparaison tout en respectant les contraintes de temps, d'énergie

consommée, de ressources utilisées et de qualité de la prédiction de l'identité. Les réseaux de neurones de type fuzzy ARTMAP (Carpenter et al., 1992) sont un bon exemple de ce type de comparateur. Ils sont capables d'effectuer un apprentissage stable et rapide de façon incrémentale à l'aide de données d'entraînement limitées, et offrent une bonne précision en généralisation (Granger et al., 2007). L'utilisation de distances de type L_1 (distance de type pâté de maison ou « Manhattan ») et d'opérateurs fuzzy ET ainsi que OU permet de prédire l'identité d'un individu rapidement et avec peu d'espace mémoire disponible. L'architecture de l'algorithme permet de le paralléliser à différents niveaux de granularité.

Bien qu'en utilisant les réseaux de neurones fuzzy ARTMAP et une stratégie d'apprentissage d'optimisation multiobjective par essaims de particules (OMOEP) (Granger, Prieur et Connolly, 2010), il soit possible de minimiser les ressources informatiques nécessaires à l'implémentation d'un système de reconnaissance des visages dans des séquences vidéo tout en maximisant son taux de classification, ce n'est qu'en choisissant judicieusement le processeur utilisé que la performance désirée peut être atteinte sans négliger la consommation énergétique requise par les applications électroniques mobiles.

Un système de reconnaissance de visages dans les séquences vidéo pour une application électronique mobile recherchée devrait être capable de capter les séquences vidéo à une cadence de 30 trames VGA (640x480 pixels) par seconde. Le taux de détection de visages de 50 % et la capacité d'identifier jusqu'à cinq visages simultanément implique de pouvoir produire une prédiction pour une trame unique en moins de treize millisecondes. Afin de produire une durée de vie utile de la pile adéquate, la puissance maximale du processeur doit se trouver sous le seuil des 5 Watts et le processeur choisi doit aussi avoir une mémoire cache suffisamment grande pour y emmagasiner le réseau de neurones. Idéalement, la taille de la mémoire cache locale associée à chaque nœud de calculs serait assez grande pour y emmagasiner une instance du réseau de neurones.

Toutefois, le choix du processeur n'est pas quelque chose de simple en raison de la consommation énergétique du processeur qui croît de façon linéaire lorsque la cadence

d'horloge du processeur augmente. Pour pouvoir tirer avantage des processeurs comportant de multiples nœuds de calculs, l'algorithme sériel fuzzy ARTMAP nécessite aussi quelques modifications pour rendre le traitement parallèle possible. Ceci implique aussi de pouvoir être capable de compiler le code source en langage C sur le processeur retenu.

Les objectifs de ce mémoire consistent à proposer et à décrire une technique pour la réalisation d'un système de reconnaissance de visages dans des séquences vidéo pour les applications électroniques mobiles partagées entre plusieurs usagers. Trois implémentations issues de cette technique seront effectuées sur des processeurs génériques commercialement disponibles et la performance de celles-ci sera évaluée en fonction de la précision de la prédiction de l'identité, les ressources matérielles nécessaires à l'implémentation, la vitesse de traitement ainsi que la consommation d'énergie du processus de prédiction de l'identité.

Les contributions de ce mémoire résident dans l'utilisation d'un réseau de neurones artificiels de type fuzzy ARTMAP, dont la performance a été optimisée à l'aide d'une nouvelle stratégie d'entraînement employant un algorithme OMOEP. Ce dernier a été modifié pour permettre le traitement des séquences vidéo et le travail en parallèle pour atteindre les objectifs de qualité de prédiction, de temps de traitement et de consommation d'énergie.

Pour y arriver, le chapitre 1 contient la théorie nécessaire pour la compréhension de la biométrie. L'extraction et la sélection de caractéristiques avec l'ACP, les réseaux de neurones fuzzy ARTMAP et l'optimisation multiobjective à l'aide d'essaims de particules sont présentées au chapitre 2. Dans le chapitre 3, l'implémentation matérielle en ce qui concerne le choix du processeur et les modifications algorithmiques nécessaires sont discutées. Dans le chapitre 4, on trouve la méthodologie expérimentale pour évaluer les performances. Finalement, le chapitre 5 présente les résultats de l'évaluation des performances ainsi qu'une analyse de ceux-ci.

CHAPITRE 1

IDENTIFICATION BIOMÉTRIQUE

Le concept primordial derrière la sécurité est de séparer les actifs des différentes menaces présentes dans leurs environnements. Pour y arriver, nous avons besoin d'un moyen d'identification précis et fiable. Ceci est primordial au point que d'apprendre à reconnaître le visage de sa mère est une des premières choses qu'un enfant apprend à faire après la naissance.

Pour une très longue période, l'identité d'une personne n'était établie que par une longue période d'interactions avec l'individu, ou bien lorsqu'un tiers dont l'identité était établie avec les deux autres individus faisait les introductions. Même si ceci fût suffisant pour le début de la civilisation humaine, la taille des regroupements humains et l'étalement de la population ont montré l'impossibilité de toujours trouver un tiers disponible pour assurer de l'identité d'un individu.

L'autre grande évolution, en ce qui concerne l'identification, fût d'utiliser des jetons d'authentification. Ces jetons peuvent être matériels, comme dans le cas de clés, cartes, lettres, sceaux ou tout autre objet identifiant la personne, ou ils peuvent aussi être immatériels dans le cas de mots de passe, poignée de main secrète, réponses à une énigme ou toute autre chose que seul l'individu peut connaître (Prabhakar, Pankanti et Jain, 2003).

Malgré les avantages que l'utilisation de jetons d'authentification apporte, ils ont aussi créé de nouveaux problèmes (de Luis-García et al., 2003). Les jetons matériels peuvent être égarés, endommagés ou bien même volés. Quant à eux, les jetons immatériels peuvent être oubliés, copiés et dans certains cas l'individu pourrait être forcé de révéler l'information jouant le rôle de jeton immatériel. Ceci a provoqué une complexification constante des jetons d'authentification qui, à chaque génération, pousse le perfectionnement à un niveau supérieur afin de déjouer ceux qui voudraient déjouer le système.

La vie moderne, avec son nombre incalculable de systèmes avec lesquels nous devons interagir sur une base quotidienne, a causé une prolifération marquée du nombre de jetons d'authentification qu'une personne doit posséder pour fonctionner normalement. Que ce soit par l'utilisation de mots de passe trop simples, la réutilisation du même mot de passe pour plusieurs systèmes, le contournement ou même la désactivation d'options de sécurité, l'utilisateur en tant que tel est devenu un risque de sécurité non négligeable (Jain, Ross et Pankanti, 2006).

La reconnaissance biométrique vise à améliorer le processus, en transformant les jetons d'authentification de quelque chose que l'utilisateur sait ou possède, en quelque chose d'intrinsèque à l'individu. Ceci peut être soit une mesure anatomique directe ou bien même une mesure comportementale de l'individu.

Malgré tous ses bénéfices, la biométrie comporte aussi sa part de problèmes (Prabhakar, Pankanti et Jain, 2003). Toutes les caractéristiques ne sont pas aussi efficaces pour l'identification d'individus ni résistantes aux tentatives de déjouer le système. Il y aussi la possibilité que le modèle biométrique soit volé, ce qui compromettrait l'identité de la personne de façon permanente.

1.1 Validation, identification et dépistage

Bien que les différentes applications de reconnaissance biométrique sont nombreuses et variées, il est possible de les regrouper selon trois grandes catégories : la validation, l'identification et l'échantillonnage (Jain, Ross et Pankanti, 2006).

Dans une application typique de validation d'identité, aussi appelé authentification, un système requiert que l'individu cible affirme son identité avant que le système vérifie si l'individu est vraiment la personne qu'il affirme être (Jain, Ross et Pankanti, 2006). Ceci est semblable au processus vécu par un individu traversant une frontière internationale et affirme son identité en présentant un passeport valide au douanier. Par après, ce dernier décide si la

personne devant lui est la personne décrite dans le passeport. Ce type d'application permet au système de seulement comparer les signaux biométriques au modèle associé avec l'identité que l'individu affirme être. Ce type de système est souvent utilisé lorsqu'on désire empêcher plusieurs individus d'utiliser la même identité (Wayman, 2001).

Dans les applications de type identification, le système tente de trouver l'identité d'un individu parmi tous ceux inscrits dans le système (Jain, Ross et Pankanti, 2006). Ce type d'application demande au système de comparer le signal biométrique contre tous les modèles d'identités. Ceci peut être utile pour empêcher une personne de tenter d'utiliser plusieurs identités (Wayman, 2001) ou bien de rendre le processus d'une application de validation d'identité plus conviviale en retirant le besoin que l'individu affirme son identité à chaque interaction (Jain, Ross et Prabhakar, 2004). Ce genre de système nécessite un classificateur efficace capable de gérer un grand débit de données avec le minimum d'interventions humaines. L'identification ne peut s'effectuer qu'avec l'utilisation de la biométrie, car les jetons d'authentification classiques ne sont pas suffisamment personnels à l'individu à identifier pour permettre de le distinguer des autres identités possibles (Jain, Ross et Pankanti, 2006).

Les applications de dépistage quant à elles sont un sous-groupe des applications d'identification, mais ayant plusieurs contraintes supplémentaires (Jain, Ross et Pankanti, 2006):

- présence d'individus inconnus par le système;
- de nouveaux individus peuvent être inscrits dans le système tout durant la phase d'opération du système et peu de données sont disponibles pour l'inscription;
- il n'y a pas de coopération de la part de l'individu qui peut aller jusqu'à vouloir volontairement tromper le système;
- le positionnement de l'individu et les conditions d'échantillonnage sont moins contrôlés;
- le système doit être capable de comparer à grande vitesse les signaux biométriques contre un nombre limité de modèles.

Tout comme les applications d'identification, le dépistage ne peut s'effectuer à l'aide de jetons d'authentification classiques et requiert l'utilisation de caractéristiques biométriques intrinsèques à l'individu (Jain, Ross et Pankanti, 2006).

1.2 Caractéristiques biométriques ainsi que leurs performances

Bien que n'importe quelle mesure physiologique ou comportementale puisse être utilisée dans un système de reconnaissance biométrique, elles ne sont pas toutes aussi performantes les unes que les autres. Ainsi, toute bonne mesure biométrique potentielle doit satisfaire les critères suivants (Jain, Ross et Prabhakar, 2004) :

- Universelle : Par ceci, nous voulons dire que la caractéristique biométrique doit être présente pour tous les individus. Par exemple, il est impossible de recueillir l'empreinte digitale d'un individu qui a été amputé de la main ou bien de mesurer la démarche d'une personne quadriplégique.
- Distinctive : La mesure effectuée sur un individu doit être suffisamment différente de celles effectuées sur les autres individus pour permettre de discriminer entre eux. Ici, on peut penser au code génétique qui varie significativement d'une personne à l'autre tandis que la taille est une mesure que plusieurs individus partagent.
- Permanente : Le fait que le corps humain vieillit implique qu'après un certain laps de temps, une mesure biométrique faite sur un individu peut être très différente de la mesure initiale utilisée pour l'inscription dans le système. Par conséquent, il est important de choisir une caractéristique biométrique qui reste stable durant la vie de l'individu. Un bon exemple de ceci est le motif présent dans l'iris ou même le code génétique de l'individu.
- Facile à mesurer : Ceci représente à quel point il est facile de recueillir et de quantifier la mesure biométrique. Il est facile d'enregistrer le son de la voix d'un individu à l'aide d'un microphone, mais il est beaucoup moins facile d'obtenir une image de bonne qualité de la rétine des individus.

- Efficace : L'efficacité fait référence à la quantité de ressources nécessaires afin d'obtenir le niveau de qualité désiré dans le temps requis. Un test sanguin chimique requiert du matériel sophistiqué et une longue période de temps avant d'obtenir un résultat comparativement à recueillir l'image du visage d'un individu à l'aide d'une caméra numérique.
- Acceptable : Ceci correspond aux aspects socioculturels de la reconnaissance biométrique. Même si avec la surabondance de caméras de sécurité, la population normale ne se soucie plus d'avoir son image recueillie par de multiples systèmes, l'idée de traverser un détecteur capable de produire une image détaillée du corps d'un individu sans vêtement en indispose plusieurs.
- Robuste : Par ceci, nous faisons référence à la difficulté de forcer le système à produire une prédiction erronée par l'utilisation d'une technique frauduleuse. L'utilisation de technique de maquillage pour effets spéciaux a déjà permis de tromper autant les humains que les machines lors de reconnaissance faciale.

Pour le reste de la section, nous allons brièvement présenter différentes caractéristiques biométriques utilisées afin d'identification. Parmi les plus communes, nous trouvons (Jain, Ross et Prabhakar, 2004):

- L'acide désoxyribonucléique (ADN) : L'ADN est considéré par certains comme étant le discriminant biométrique idéal. Il consiste en la séquence des acides aminés dans le code génétique d'un individu qui lui est unique à l'exception de vrais jumeaux. Bien que très présent en criminalistique, plusieurs problèmes empêchent le déploiement à grande échelle. Le besoin d'utiliser de l'équipement spécialisé manipulé par du personnel expérimenté, de longs délais avant d'obtenir les résultats, la susceptibilité à la contamination et le risque que l'ADN soit utilisé pour discriminer un certain sous-groupe de la population sont parmi les exemples de problèmes qui bloquent l'utilisation de l'ADN comme caractéristique biométrique pour la reconnaissance automatique.

- La démarche : La démarche d'un individu est une séquence de mouvements complexes qui peut être utilisée pour l'identifier. Elle n'est pas très distinctive et ne reste pas constante dans le temps, mais elle est facile à implémenter et socialement très acceptable.
- Les empreintes digitales : Les empreintes digitales sont utilisées à des fins d'identification depuis plus d'un siècle. Elles sont produites par les structures cutanées recouvrant les doigts et sont développées lors de la croissance fœtale. Elles sont uniques à l'individu et très acceptables socialement à être utilisées pour des fins d'identification, mais doivent généralement être recueillies par contact. Il y a aussi une certaine partie de la population qui se voit incapable de produire une empreinte digitale mesurable.
- L'iris : L'iris est l'anneau de couleur entourant la pupille de l'œil et gère la quantité de lumière qui pénètre à l'intérieur du globe oculaire. La texture des tissus de l'iris est très distinctive et diffère même dans le cas de vrais jumeaux. Elle est très difficile à altérée chirurgicalement et l'utilisation d'accessoires pour changer son apparence est facilement détectable. L'acquisition du signal biométrique nécessite tout de même la coopération de l'individu cible.
- La main : La géométrie de la main peut être utilisée pour des fins de reconnaissance en utilisant une série de mesures des sous-structures de la main de l'individu. La structure de la main semble être plus constante au fil du temps que son apparence. L'implémentation est relativement simple et peu coûteuse, mais n'est probablement pas assez distinctive pour différencier un individu d'une grande population.
- La rétine : Le complexe réseau vasculaire compris dans la rétine humaine est peut-être la structure la plus distinctive du corps humain et il est extrêmement difficile de la modifier chirurgicalement. La difficulté d'acquérir une image de qualité de la rétine nécessite l'entière coopération de l'individu cible.
- La signature : La façon qu'une personne signe son nom est depuis longtemps une caractéristique biométrique utilisée pour la validation de l'identité d'une personne. Une signature a un poids légal dans la majorité des pays. Comme beaucoup de caractéristiques comportementales, elles sont sujettes à varier en fonction de l'état émotionnel de l'individu. La contrefaçon de signature est quelque chose de très ancien et les nouveaux

moyens technologiques disponibles pour les faussaires rendent leur détection de plus en plus ardue.

- Le système sanguin : La signature thermique du système sanguin humain peut être utilisée à des fins de reconnaissance biométrique. Ceci requiert l'utilisation de caméras infrarouges qui sont plus dispendieuses que les caméras classiques et peuvent être difficiles à opérer dans des environnements non contraints.
- Le visage : La reconnaissance faciale est une des caractéristiques biométriques les plus faciles à recueillir en utilisant une caméra. La reconnaissance peut se faire soit sur une base statique (image par image) ou bien sur une base dynamique (avec des séquences vidéo). Il reste tout de même des doutes sur l'utilisation d'un visage comme élément distinctif pour identifier un individu parmi une grande population (Golfarelli, Maio et Malton, 1997).
- La voix : La voix humaine est une caractéristique biométrique intéressante, puisqu'elle dépend de la structure anatomique de l'individu ainsi que de l'apprentissage du langage fait lors de l'enfance. La capture de la voix est relativement facile à effectuer, à l'aide d'un microphone, mais elle est susceptible à être corrompue par les bruits ambiants.

1.3 Système typique de reconnaissance de visages dans des séquences vidéo

Pour un système automatique de reconnaissance du visage dans des séquences vidéo générique (Figure 1.1) moderne, nous pouvons travailler avec l'hypothèse que la scène où se trouvent les individus cibles soit couverte par une ou plusieurs caméras numériques. Chacune de ces caméras fait l'acquisition d'une série de trames vidéo donnant ainsi au système de multiples images 2D des individus présents dans la scène. La première étape consiste à effectuer la détection du visage et la segmentation des images pour produire des régions d'intérêt (RI) correspondant aux différents visages dans chaque trame. Après l'obtention d'une RI de la qualité requise, des caractéristiques discriminantes et invariantes sont extraites et sélectionnées avant d'être composées en deux motifs d'entrée, *a* et *b*, aux fins de classification et de suivi respectivement.

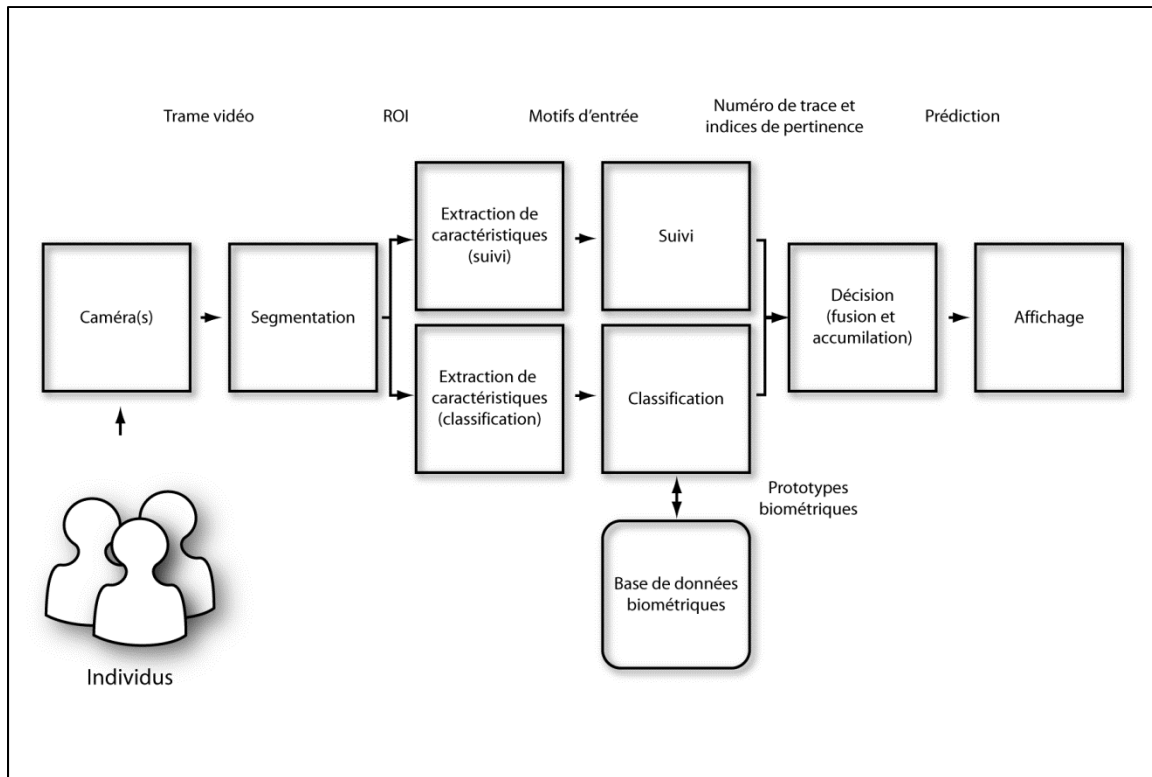


Figure 1.1 Système typique de reconnaissance de visage dans des séquences vidéo

Lors de l'inscription dans le système, un ou plusieurs motifs d'entrée, \mathbf{a} , d'un même individu sont utilisés pour créer un modèle ou prototype facial de celui-ci avant d'être stocké dans la base de données afin d'être utilisé pour la prédiction de l'identité. La prédiction est généralement générée à l'aide d'un classificateur entraîné pour associer les motifs d'entrée à une des L classes d'identité associées aux individus inscrits dans le système. Ainsi, lors de la phase opérationnelle, les motifs d'entrée \mathbf{a} sont comparés aux différents prototypes du système pour produire un indice de probabilité $\Phi_i(\mathbf{a})$ pour chacun des L individus et sont comparés au seuil γ_i afin de produire une décision spécifique à l'application. Dans une application de type vérification, on accepte ou rejette une affirmation d'identité tandis que les applications de type identification et surveillance doivent trouver l'identité la plus plausible parmi toutes celles inscrites dans le système. Pour réduire l'ambiguïté lors de la décision, certaines caractéristiques spatiotemporelles sont assemblées dans le motif \mathbf{b} qui est utilisé pour effectuer le suivi de l'individu dans une série de RI consécutives.

Les approches image/vidéo utilisent une ou plusieurs RI provenant d'images statiques de référence pour créer les prototypes faciaux. Les approches vidéo/vidéo quant à elles utilisent des séquences vidéo pour créer leurs prototypes faciaux. Par exemple, dans le domaine de la surveillance vidéo, un analyste peut décider d'inscrire un ou plusieurs individus dans le système à l'aide d'une séquence vidéo issue d'une caméra de sécurité pour ensuite suivre et classer leurs activités captées à l'aide d'autres caméras reliées au système.

1.4 État de l'art en reconnaissance de visages dans des séquences vidéo

Avec l'intérêt que la reconnaissance automatique des visages a suscité au cours des dernières années, beaucoup de nouvelles techniques et approches ont été proposées (Zhao et al., 2003) (Abate et al., 2007) (Matta et Dugelay, 2009) (Chellappa, Bicego et Turaga, 2009). Ces techniques peuvent généralement être divisées en deux catégories. La première regroupe les approches utilisant les images photographiques pour l'entraînement et pour la phase opérationnelle, tandis que la deuxième, quant à elle, concerne celles utilisant les séquences vidéo pour les mêmes tâches.

Dans la première catégorie, celle utilisant les images photographiques, on retrouve les premières approches proposées pour la reconnaissance automatique de visages. Les approches de type globales telles *Eigenfaces* (Sirovich et Kirby, 1987) (Turk et Pentland, 1991) basée sur l'analyse des composantes principales (ACP), *Fisherfaces* (Belhumeur, Hespanha et Kriegman, 1997) basée sur l'analyse du discriminant linéaire (ADL) et la technique basée sur l'analyse des composantes indépendantes (ACI) (Bartlett, Movellan et Sejnowski, 2002), utilisent la totalité de l'information contenue dans les valeurs des pixels de l'image pour identifier les individus. L'approche de type locale utilisant la méthode d'appareillement de graphes élastiques (AGE) (Wiskott et al., 1997) utilise le positionnement des traits du visage pour l'identification.

La capacité des réseaux de neurones artificiels de trouver des solutions dans des problèmes non linéaires de reconnaissances de formes a aussi été exploitée pour la reconnaissance de

visages (Abate et al., 2007). Plusieurs différentes architectures de réseaux de neurones artificiels ont été utilisées pour la reconnaissance de visage (Abate et al., 2007) soient les cartes autoassociatives (CAA), les réseaux de neurones de décision basée sur la probabilité (RNBDP), les réseaux de neurones à bases radiales (RNBR) ainsi que les réseaux de neurones fuzzy ARTMAP (Granger, Prieur et Connolly, 2010). Pour augmenter la performance des réseaux de neurones artificiels, l'utilisation d'algorithmes d'optimisation évolutive est un outil très puissant (Yao, 1999). Dans le cas spécifique des réseaux de neurones de type fuzzy ARTMAP, l'algorithme d'optimisation par essaim de particules s'est révélé un outil approprié pour maximiser le taux de classification (Granger et al., 2007). La dégradation de la performance en compression du réseau fuzzy ARTMAP optimisé à l'aide d'essaims de particules peut être mieux gérée en utilisant un algorithme d'optimisation évolutive multiobjective (Granger, Prieur et Connolly, 2010). Cette performance en compression est importante, car elle est directement liée à l'espace mémoire occupé par le réseau de neurones ainsi qu'à la vitesse d'exécution de l'algorithme et la consommation d'énergie du système. L'algorithme d'optimisation évolutive multiobjective par essaim de particules (OMOEP) (Coello, Pulido et Lechuga, 2004) fait partie de la catégorie des algorithmes hybrides dans la grande famille des algorithmes d'optimisation évolutive multiobjective (Zhou et al., 2011).

La deuxième catégorie, celle utilisant les séquences vidéo, peut à son tour être subdivisée en deux sous-catégories (Chellappa, Bicego et Turaga, 2009) : celle regroupant les techniques utilisant les séquences vidéo pour améliorer la performance d'une technique de type image photographique et celle regroupant les techniques qui utilisent l'information spatio-temporelle aux fins de reconnaissance de visages.

Il existe plusieurs façons d'améliorer les approches utilisant les images photographiques pour la reconnaissance de visages avec l'utilisation de séquences vidéo. Il est possible d'améliorer la qualité de la représentation du visage en utilisant les variations de pose et d'illumination que les différentes trames de la séquence vidéo offrent afin de créer un sous-espace mathématique de représentation du visage d'un individu (Lee et al., 2005) (Arandjelovic, Ar

et Cipolla, 2006) (Stallkamp, Ekenel et Stiefelhagen, 2007). Il est aussi possible d'améliorer la localisation du visage en effectuant le suivi des traits du visage dans le temps pour en extraire le positionnement 3D du visage (Jebara et Pentland, 1997) (Chan et Vasconcelos, 2005) (Aggarwal, Veeraraghavan et Chellappa, 2005). Une autre possibilité est d'améliorer la performance du classificateur en fusionnant l'information (Ross, Nandakumar et Jain, 2006) provenant de toutes les trames de la séquence vidéo. La technique proposée dans cet ouvrage se trouve placée dans cette sous-catégorie.

D'autres approches utilisent de façon plus intrinsèque l'information spatiotemporelle qui se trouve dans les séquences vidéo. L'utilisation de filtres à particules pour établir un suivi et une classification simultanée (Li et Chellappa, 2001) (Zhou, Krueger et Chellappa, 2003) est un exemple de ce type d'approche qui utilise le suivi d'éléments faciaux propres aux individus inscrits dans le système pour trouver quel individu possède les éléments faciaux qui offrent le meilleur suivi. Une autre approche est d'utiliser un modèle caché de Markov (MCM) pour chacun des individus inscrits dans le système de reconnaissance afin d'analyser l'information temporelle contenue dans la séquence vidéo (Hadid et Pietikainen, 2005) (Xiaoming et Tsuhan, 2003) (Tistarelli, Bicego et Grosso, 2009). Les expressions et les mouvements involontaires du visage extraits de la séquence vidéo peuvent aussi être utilisés pour des fins de reconnaissance (Chellappa, Bicego et Turaga, 2009).

Dû aux fortes contraintes de conception qu'impose la reconnaissance de visages, le matériel utilisé a principalement été des circuits intégrés spécifiques à l'application (CISA) et processeurs spécialisés. Des systèmes de reconnaissance de visages ont déjà été implémentés en utilisant l'intégration de systèmes à grande échelle (ITGE) (Nagel et al., 2002), des réseaux prédiffusés programmables par l'utilisateur (RPRU) (Borgatti et al., 2003), des processeurs de traitement numérique de signal (PSN) (Batur, Flinchbaugh et Hayes, 2003) (Mao et Bigdeli, 2004), des processeurs embarqués (Kondo et al., 2005) ou spécialisés pour le traitement vidéo (Kleihorst et al., 2007) afin d'obtenir une haute cadence de traitement avec la précision nécessaire.

Par contre, même s'ils ne sont toujours pas capables d'offrir le même niveau de performance, les processeurs génériques restent toujours une possibilité intéressante lorsque nous tenons compte des coûts supplémentaires que l'implémentation et la maintenance d'un système utilisant du matériel si spécialisé requièrent ainsi que la réduction du temps nécessaire pour commercialiser le produit (Reyneri, 2003). Ceci est d'autant plus vrai lorsque nous considérons les nouvelles générations de processeurs à basse puissance qui brouillent la distinction entre les processeurs génériques et les processeurs embarqués. Les réseaux de neurones artificiels de type fuzzy ARTMAP ont eux-mêmes été implémentés sur divers types de plateforme matérielle. L'algorithme a été utilisé sur un superordinateur (Malkani et Vassiliadis, 1995), sur des grappes de calculs (Castro et al., 2007), dans une implémentation ITGE (Lubkin et Cauwenberghs, 1999) et même en optoélectronique (Blume et Esener, 1997).

Ainsi, l'approche présentée dans cet ouvrage se concentre sur l'identification biométrique pour le contrôle d'accès dans des applications d'appareils électroniques mobiles ou distribués en utilisant les séquences vidéo pour améliorer la performance du classificateur neuronique fuzzy ARTMAP entraîné pour la reconnaissance de visages à l'aide de la stratégie d'apprentissage d'optimisation multiobjective par essaims de particules et implémenté sur un processeur multicœur à faible consommation énergétique.

1.5 Défis à la réalisation

Les grands défis dans la conception d'un système de reconnaissance des visages dans des séquences vidéo sont la véracité des prédictions, les ressources mémoires nécessaires, la vitesse d'exécution ainsi que la consommation énergétique.

Puisque le visage humain est un objet flexible et déformable, la tâche de correctement modéliser le prototype d'identité d'un individu est très complexe. En plus des variations dans l'expression faciale des individus, la véracité des prédictions est aussi affectée par les variations de positionnement et les conditions d'illumination (Abate et al., 2007). L'usage de

produits cosmétiques et le processus de vieillissement sont aussi une des sources de variation intra classe.

Le processus d'entraînement du classificateur adaptatif qui produira les prédictions d'identité est souvent rendu difficile par le peu de données disponibles lors de l'enrôlement d'un individu dans le système. Il n'est pas rare de n'avoir qu'une courte séquence vidéo ou même qu'un faible nombre d'images statiques disponibles pour l'entraînement. De plus, la faible résolution des caméras de sécurité peut aussi être une source de problème pour le système.

Le défi lié à la gestion des ressources mémoires disponibles vient de la nécessité d'entreposer et d'accéder rapidement aux prototypes d'identité créés lors de l'enrôlement. La taille de la base donnée contenant les prototypes n'influence pas uniquement la quantité de mémoire nécessaire à son stockage, mais aussi la véracité des prédictions ainsi que la vitesse de traitement du système. Ceci est encore plus important dans des applications mobiles ou distribuées, puisque ce type d'application possède des contraintes matérielles beaucoup plus sévères qu'un poste informatique commun.

Dans une application de reconnaissance de visage dans des images statiques, la vitesse à laquelle le système produit une prédiction n'a comme contrainte que d'être suffisante pour que la prédiction soit produite avant que l'information ne soit plus pertinente. Ceci peut aller de secondes dans les cas de contrôle d'accès jusqu'à des semaines lors d'identification criminalistique. Avec la reconnaissance de visages dans des séquences vidéo, la contrainte supplémentaire est d'effectuer la prédiction avant que la prochaine région d'intérêt soit soumise au système. Dans la plupart des cas, ceci implique une cadence de traitement d'au moins trente (30) régions d'intérêt par secondes par individu présent dans la scène. Sans une vitesse de traitement proche du temps réel, la quantité de régions d'intérêt en attente de traitement ne cesserait de croître jusqu'à ce que l'espace pour les conserver vienne à manquer.

La consommation d'énergie est un aspect de la reconnaissance de visage dans des séquences vidéo qui est rarement traité dans la littérature. Mais avec l'émergence des applications

mobiles et distribuées, la consommation d'énergie ne peut être ignorée dû à son lien direct avec la longévité de l'utilisation d'un appareil alimenté à l'aide d'une pile. La consommation d'énergie est généralement inversement proportionnelle à la vitesse de traitement et la véracité des prédictions.

1.6 Conclusion

Dans ce chapitre, il a été possible de voir que la reconnaissance biométrique remplace les témoins traditionnellement utilisés pour établir l'identité d'un individu par quelque chose d'intrinsèque à celui-ci. Les applications biométriques peuvent être divisées en trois groupes : la validation, l'identification et le dépistage. Les caractéristiques biométriques peuvent être physiologiques ou bien comportementales et leur performance est évaluée selon leur universalité, leur caractère distinctif, leur permanence, leur efficacité, leur acceptabilité sociale et leur robustesse.

Il a aussi été possible de voir que l'approche proposée dans cet ouvrage se concentre sur les applications d'identification pour le contrôle d'accès pour des appareils électroniques mobiles ou distribués en utilisant des séquences vidéo pour augmenter la performance du classificateur neuronique fuzzy ARTMAP entraîné à l'aide d'une stratégie d'apprentissage OMOEP et implémenté sur un processeur multicœur générique à basse puissance. Les défis principaux à la réalisation d'un système de reconnaissance de visages sont le peu de ressources matérielles disponibles, le besoin de minimiser la consommation énergétique pour maximiser la durée de vie de la pile, un traitement des données près du temps réel tout en maintenant une bonne probabilité de succès en prédiction de l'identité des individus.

Comme il a été montré dans le système générique de reconnaissance de visages dans des séquences vidéo, l'extraction et la sélection de caractéristiques ainsi que la production de la prédiction de l'identité de l'individu cible sont le cœur du système. Par conséquent, le prochain chapitre explorera plus en détail leur fonctionnement.

CHAPITRE 2

PRÉDICTION DE L'IDENTITÉ

Une fois qu'un visage a été détecté par le système et qu'une RI a été produite, une série d'opérations doivent être effectuées avant de pouvoir produire une prédiction de l'identité de l'individu cible. Ainsi, dans ce chapitre, il sera possible d'étudier avec plus de minutie l'extraction et la sélection de caractéristiques, les classificateurs neuroniques de type fuzzy ARTMAP et leurs stratégies d'apprentissage par optimisation évolutionnaire.

2.1 Extraction et sélection de caractéristiques avec l'analyse par composantes principales

Il existe plusieurs types de caractéristiques visuelles pouvant être utilisées pour l'identification biométrique. Elles peuvent généralement être regroupées en quatre catégories : les caractéristiques géométriques, les caractéristiques fréquentielles, basées sur les textures ainsi que celles basées sur les projections.

Les caractéristiques géométriques regroupent les différentes mesures telles la hauteur ou la largeur mesurée à différents points, les mesures sur le contour du visage et l'utilisation de modèles déformables. Le grand désavantage des techniques géométriques est la nécessité de localiser précisément un grand nombre de points sur l'image afin de pouvoir effectuer les mesures (Kung, Mak et Lin, 2005). Le contenu fréquentiel de l'image peut lui aussi être exploité pour l'extraction de caractéristiques. L'information haute fréquence est extraite en utilisant des techniques telles la détection d'arrêtes (Canny, Sobel, Prewit, etc.) ou l'utilisation du Laplacien du gaussien. Les techniques telles que les filtres passe-bas, la diminution de la résolution et la structure de la pyramide gaussienne peuvent être utilisées pour extraire les caractéristiques basse fréquences de l'image (Kung, Mak et Lin, 2005). L'information contenue dans la texture de l'image est aussi utile à l'identification biométrique (Kung, Mak et Lin, 2005). Les techniques telles l'utilisation d'ondelettes de

Gabor et les patrons locaux binaires (Ahonen, Hadid et Pietikainen, 2006) permettent d'extraire l'information texturale en caractéristiques utiles. Les technique de projection telles l'analyse par composantes principales, l'analyse des discriminants linéaires et l'analyse des composantes indépendantes permettent d'extraire des caractéristiques en transposant les données dans un sous-espace plus représentatif (Kung, Mak et Lin, 2005). Ce type de projection permet aussi de réduire la dimensionnalité des motifs présentés à l'entrée du classificateur.

L'analyse par composantes principales (ACP), parfois nommée comme transformée de Karhunen-Loève, est une technique statistique de projection linéaire qui tente de transposer les données dans un espace plus représentatif. Pour arriver à ceci, le référentiel est transformé de façon à aligner les axes dans la direction de la variance maximale.

Cette analyse se trouve au cœur de la technique de reconnaissance de visage *Eigenfaces* (Sirovich et Kirby, 1987; Turk et Pentland, 1991). Même si cette approche comporte plusieurs faiblesses, sa simplicité de compréhension et d'implémentation ainsi qu'un bon niveau de performance maintiennent la pertinence de l'ACP jusqu'à présent.

L'avantage principal de l'ACP est la possibilité de réduction de dimensionnalité qu'elle offre. Le sous-espace ne peut jamais parfaitement représenter toutes les subtilités présentes dans l'espace original. Ceci se voit lorsque nous comparons le taux de classification d'un classificateur utilisant toute l'information initiale à celui qui utilise les données transposées même en gardant toutes les composantes ($k = d$). Par contre, la très grande dimensionnalité des informations contenues dans les images utilisées en reconnaissance de visage fait que l'ordre de grandeur de la réduction de dimension demeure largement supérieur à la perte de précision qu'elle cause.

Bien que l'ACP maximise la rétention d'information, rien ne garantit la pertinence de cette information pour des fins de classification. En effet, la variance maximale, sur laquelle vient s'aligner l'axe de la composante principale, peut simplement être le produit d'un bruit dans le

signal. C'est pourquoi d'autres voies ont été explorées, notamment l'analyse des discriminants linéaires et l'analyse des composantes indépendantes.

L'analyse des discriminants linéaires (ADL) tente de résoudre le problème de la non-garantie de la pertinence en recherchant une représentation maximisant la discrimination des différentes classes du problème. Par contre, elle offre moins de flexibilité pour la réduction de dimensionnalité puisque le nombre de composantes présentes dans le sous-espace est égal au nombre de classes. La performance d'ADL comparativement à celle de l'ACP devrait théoriquement être supérieure, mais en réalité l'ACP arrive tout de même à produire une meilleure performance en classification que l'ADL (Belhumeur, Hespanha et Kriegman, 1997; Martinez et Kak, 2001).

L'analyse des composantes indépendantes (ACI) offre la possibilité de contourner une autre des limitations théoriques de l'ACP, c'est-à-dire, l'obligation de produire un sous-espace dont chacun des axes est perpendiculaire aux autres. En maximisant l'indépendance des composantes, l'ACI permet d'exploiter des relations inter pixels à un plus haut niveau que l'ACP (Bartlett, Movellan et Sejnowski, 2002). Par contre, il reste toujours un doute sur la performance réelle sur des problèmes de reconnaissance de visages d'ACI relativement à celle de l'ACP dû à des résultats conflictuels (Baek et al., 2002).

Puisque l'analyse par composantes principales est influencée par la position du visage, une étape de prétraitement est nécessaire. Au cours de celle-ci, une rotation est effectuée pour aligner horizontalement la droite traversant les deux yeux. Une fois ceci accompli, une translation est accomplie pour distribuer l'espace entre les yeux uniformément de part et autre de la ligne médiane de l'image. Une dernière translation positionne verticalement la droite traversant les yeux pour se situer à la valeur moyenne de toutes les images de l'ensemble de données et complète le prétraitement.

Pour appliquer l'ACP (Algorithme 2.1) à un ensemble de données X composé d'un nombre n de vecteurs à d dimensions, la première étape consiste à calculer le vecteur moyen μ de X

ainsi que la matrice de covariance $\mathbf{\Omega}$ de dimension $d \times d$. Les vecteurs propres \mathbf{E} et les valeurs propres associées λ de la matrice de covariance $\mathbf{\Omega}$ sont ensuite calculés et sont triés de façon décroissante de valeurs propres.

Algorithme 2.1 Analyse par composantes principales

A. Prétraitement:

Aligner les yeux avec l'horizontal pour chacune des RI
 Centrer les yeux horizontalement pour chacune des RI
 Aligner les yeux verticalement sur la position moyenne de toutes les RI

B. Analyse:

$\mu = \text{moyenne}(\mathbf{X})$
 $\mathbf{\Omega} = \text{covariance}(\mathbf{X})$
 $\mathbf{E} = \text{vecteurs propres}(\mathbf{\Omega})$
 $\lambda = \text{valeurs propres}(\mathbf{\Omega})$
 $\mathbf{E}' = \text{ordonner-selon}(\mathbf{E}, \lambda)$
 $\mathbf{X}' = \mathbf{B} \cdot \mathbf{X}$

C. Post-traitement:

Convertir les valeurs normalisées en valeurs proportionnelles : $[0,1]$

En choisissant uniquement les k premiers vecteurs de \mathbf{E} , on peut créer une matrice de transformation \mathbf{B} de $d \times k$ dimensions, permettant de projeter les données de \mathbf{X} dans un sous-espace à k dimensions. Le nombre de vecteurs k peut être choisi de plusieurs façons. Dans certains cas, il est choisi en vue d'obtenir une proportion précise de l'information initiale, les vecteurs dont la somme de leurs variances représente 95 % de la variance totale par exemple. Dans d'autres cas, il peut être simplement choisi de façon à maximiser le taux de classification d'un système de reconnaissance de formes.

Une fois la projection terminée, les valeurs sont converties pour être proportionnellement contenues entre zéro et un, afin de pouvoir être soumises comme motifs d'entrée au réseau de neurones fuzzy ARTMAP.

2.2 Classificateur neuronique fuzzy ARTMAP

En reconnaissance de forme, le rôle du classificateur est de choisir à laquelle des catégories connues appartient le motif, sous forme de vecteur de caractéristiques, qui lui est présenté à son entrée (Duda, Hart et Stork, 2001). Les classificateurs statistiques de type « génératif » tentent de créer des modèles statistiques pour représenter les différentes catégories. Les classificateurs de type Bayesiens et les modèles de mixtures gaussiennes sont de bons exemples de classificateurs génératifs (Duda, Hart et Stork, 2001). Ce type de classificateurs est intéressant car les modèles statistiques utilisés pour représenter la fonction de distribution de probabilités sont très compacts. Par contre, les données du monde réel ne répondent pas toujours aux lois gérant ces différents modèles statistiques. En opposition, les classificateurs statistiques de type « discriminatif » tentent de diviser l'espace des possibilités de façon à identifier les différents sous-espaces que représente chacune des catégories. Parmi les classificateurs discriminatifs, nous trouvons les fonctions discriminantes linéaires ou quadratiques ainsi que les machines à vecteurs de support (Duda, Hart et Stork, 2001). En divisant l'espace des possibilités, les classificateurs discriminatifs offrent une bonne capacité de généraliser avec peu de données d'entraînement, mais l'établissement d'une frontière dans l'espace des possibilités peut être complexe lorsque la distribution de multiples catégories se chevauchant.

Il existe aussi des classificateurs statistiques utilisant des techniques non-paramétriques pour représenter la distribution de données. Les classificateurs de type « k » plus proches voisins et fenêtrage de Parzen comptent parmi les techniques non-paramétriques les plus utilisées (Duda, Hart et Stork, 2001). Ce type de classificateurs offrent généralement les meilleurs taux de classification, mais ils nécessitent de conserver en mémoire une grande partie de l'information connue pour représenter la distribution des catégories. Il est également possible d'utiliser un réseau de neurones artificiels pour effectuer la classification des motifs d'entrée en lui laissant le soin de modéliser la distribution à l'aide des connexions synaptiques entre les neurones. Les réseaux de neurones de type perceptrons multicouches (Duda, Hart et Stork, 2001), les réseaux de neurones à bases radiales (Duda, Hart et Stork, 2001) et les

réseaux fuzzy ARTMAP (Carpenter et al., 1992) sont tous des classificateurs neuroniques utilisés pour différentes applications de reconnaissance de formes.

Le choix du réseau de neurones fuzzy ARTMAP pour associer l'identité d'un individu à une RI en comparant le signal d'entrée aux modèles faciaux est dû à sa capacité d'effectuer un apprentissage stable et rapide de façon incrémentale à l'aide de données d'entraînement limitées et capables d'une bonne précision en généralisation (Granger et al., 2007). En exploitant les distances de type L_1 (distance de type pâté de maisons ou « Manhattan ») et les opérateurs *fuzzy ET* ainsi que *fuzzy OU*, la prédiction de l'identité est rapide. Son architecture permet aussi de paralléliser l'algorithme à différents niveaux de granularité.

2.2.1 Architecture

Le réseau fuzzy ARTMAP est composé de trois couches de neurones distinctes (Figure 2.1): La couche F_1 joue deux rôles distincts dans le réseau. Dans un premier temps, elle sert à projeter la forme complétée de celle soumise à l'entrée du réseau vers la couche F_2 . Dans un deuxième temps, la couche F_1 reçoit l'image du prototype associé au neurone de la couche F_2 ayant gagné la compétition pour la comparer à la forme complétée. Les neurones des couches F_1 et F_2 sont complètement interreliés au travers d'une matrice de connexions synaptiques $\mathbf{W}_1 = \{w_{ij} \in [0, 1] | i=1, 2, \dots, M; j=1, 2, \dots, N\}$.

La couche F_2 est reliée à la couche de neurones F_{AB} de grandeur égale au nombre de classes à identifier L à l'aide de la matrice de connexions synaptiques $\mathbf{W}_{AB} = \{w'_{jk} \in [0, 1] | j=1, 2, \dots, N; k=1, 2, \dots, L\}$. La couche F_{AB} représente la classe vers laquelle le neurone de la couche F_2 est associé. Toutes les connexions synaptiques de la matrice \mathbf{W}_{AB} sont égales à zéro, sauf celle qui lit le neurone de la couche F_2 au neurone de la couche F_{AB} représentant la classe associée au vecteur d'apprentissage initial qui a engendré la création du neurone de la couche F_2 .

Le vecteur d'apprentissage Y , indique la classe réelle associée à la forme soumise en entrée a et elle est fournie à la mémoire associative lors de l'apprentissage. Le vecteur de sortie X produit par la couche F_{AB} représente la prédiction de classe pour le vecteur d'entrée a .

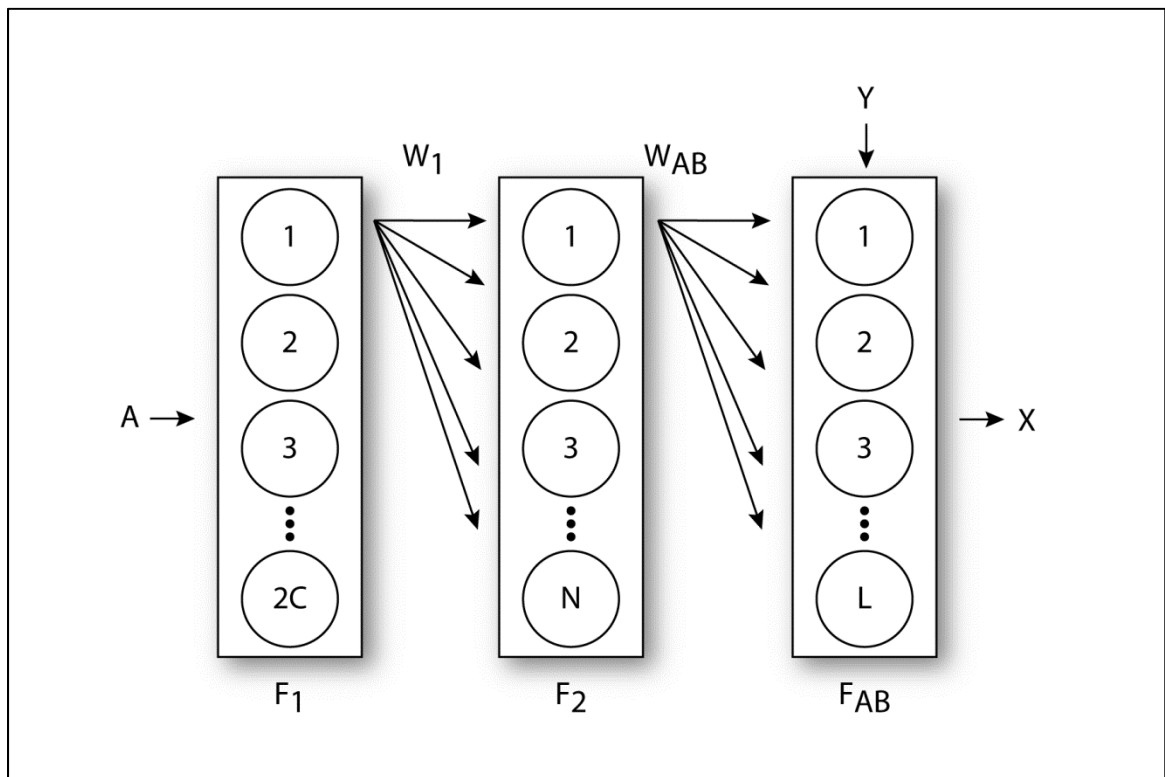


Figure 2.1 Réseau de neurones artificiels fuzzy ARTMAP

2.2.2 Hyperparamètres

La mécanique d'apprentissage de l'algorithme fuzzy ARTMAP est guidée par quatre hyperparamètres : ceux du signal, d'apprentissage, de suivi et de vigilance.

L'hyperparamètre du signal ($\alpha \in]0, \infty$) est impliqué dans la fonction de choix. En minimisant sa valeur, on obtient la « limite conservatrice » qui tend à limiter le recodage des neurones (Carpenter et al., 1992).

L'hyperparamètre d'apprentissage ($\beta \in [0, 1]$) est impliqué dans la fonction d'apprentissage. Il gère la vitesse à laquelle les valeurs des poids synaptiques d'un neurone sont modifiées lors de l'apprentissage.

L'hyperparamètre de suivi ($\varepsilon \in]0, \infty$) est impliqué dans le suivi du choix. Il est l'incrément par lequel ce seuil de vigilance se voit modifié lorsqu'une prédiction est associée à la mauvaise classe.

L'hyperparamètre de vigilance ($\rho_0 \in [0, 1]$) est impliqué dans la vérification du choix. Il représente le seuil initial de différence entre le motif d'entrée et le prototype du neurone ayant gagné la compétition pour que la prédiction soit acceptée.

2.2.3 Algorithme

Les phases d'opérations du réseau fuzzy ARTMAP sont les suivantes :

- 1) Initialisation : Remise à zéro des sorties de neurones et annulation des mécanismes d'inhibition.
- 2) Présentation de la forme d'entrée : Le vecteur d'entrée \mathbf{a} est concaténé avec son complément pour produire le vecteur \mathbf{A} .
- 3) Projection avant : Projection du vecteur \mathbf{A} sur la couche F_2 à travers les connexions synaptiques \mathbf{W}_1 .
- 4) Compétition : Le neurone de la couche F_2 ayant la valeur d'activation (T_j) la plus élevée remporte la compétition.

$$T_j(\mathbf{A}) = |\mathbf{A} \wedge \mathbf{w}_j| / (\alpha + |\mathbf{w}_j|) \quad (2.1)$$

- 5) Projection arrière : Projection du prototype de F_2 sur la couche F_1 à travers les connexions synaptiques \mathbf{W}_1 .

- 6) Comparaison : Comparaison entre le vecteur A et le prototype de la classe activée. Si la comparaison ne passe pas le test de vigilance, le neurone actif est inhibé avant de retourner à l'étape trois. Si tous les neurones actifs sont inhibés, on active un neurone et on effectue l'apprentissage du vecteur d'entrée avant de retourner à l'étape 1.

$$|A \wedge w_j| / (2c) \geq \rho \quad (2.2)$$

- 7) Prédiction : Si la comparaison passe le test de vigilance, le vecteur de la couche F_2 est projeté sur la mémoire associative qui produit le vecteur de sortie X .

En mode apprentissage, si le vecteur de sortie X n'est pas équivalent au vecteur d'apprentissage Y , le neurone actif est inhibé et le seuil du test de vigilance est modifié avant de retourner à l'étape 3.

$$\rho = |A \wedge w_j| / (2c) + \varepsilon \quad (2.3)$$

Si le vecteur de sortie X correspond au vecteur d'apprentissage Y , les poids synaptiques du neurone actif sont modifiés pour se rapprocher de la forme d'entrée en fonction du coefficient d'apprentissage.

$$w_j = \beta(A \wedge w_j) + (1-\beta)w_j \quad (2.4)$$

Dans le cas où tous les neurones sont désactivés, un nouveau neurone est créé sur la couche F_2 .

$$w_{AB} = Y \quad (2.5)$$

$$w = \beta(A \wedge w) + (1-\beta)w \quad (2.6)$$

La séquence d'apprentissage du réseau de neurones fuzzy ARTMAP est aussi présentée en forme de pseudo-code ci-dessous (Algorithme 2.2).

Algorithme 2.2 Séquence d'apprentissage du réseau de neurones fuzzy ARTMAP

```

 $q = 0$ 
tantque  $q \leq n$  faire
    Remise à zéro d'hyperparamètre de vigilance :  $\rho = \rho_0$ 
    Remise à zéro de la sortie des neurones :  $T_j = 0$ 
    Réactiver les neurones inhibés
    Concaténer le motif d'entrée  $\mathbf{a}$  avec son complément pour produire le motif  $\mathbf{A}$ 
     $match = \text{faux}$ 
    tantque  $match == \text{faux}$  faire
        Projection du motif  $\mathbf{A}$  sur la couche de neurone  $\mathbf{F}_2$  à travers les connexions
        synaptiques  $\mathbf{W}_1$  :
            
$$T_j(\mathbf{A}) = |\mathbf{A} \wedge \mathbf{w}_j| / (\alpha + |\mathbf{w}_j|)$$

        Compétition entre les neurones pour trouver la valeur d'activation la plus
        élevée parmi les neurones non inhibés :
            
$$J = \max\{T_j : j=1, \dots, f\}$$

        si  $(|\mathbf{A} \wedge \mathbf{w}_J| / (2k) \geq \rho)$  alors
            si  $(X == Y)$  alors
                
$$\mathbf{w}_J = \beta(\mathbf{A} \wedge \mathbf{w}_J) + (1-\beta)\mathbf{w}_J$$

                 $match = \text{vrai}$ 
            sinon
                
$$\rho = |\mathbf{A} \wedge \mathbf{w}_J| / (2c) + \varepsilon$$

                inhiber le neurone  $J$ 
            fin
        fin
        si (tous les neurones sont inhibés) alors
            activer un nouveau neurone :
                
$$f = f + 1$$

                
$$\mathbf{w}_{AB} = \mathbf{Y}$$

                
$$\mathbf{w} = \beta(\mathbf{A} \wedge \mathbf{w}) + (1-\beta)\mathbf{w}$$

            fin
        fin
     $q = q + 1$ 
fin

```

2.2.4 Analyse de complexité

Puisque le calcul de la valeur d'activation et du test de vigilance est la partie de l'algorithme du fuzzy ARTMAP qui nécessite le plus de calculs, il est intéressant de modéliser le coût computationnel de l'algorithme en fonction de cette étape.

Ainsi, pour chacun des n motifs de la base de données, le système doit comparer un à un les $2k$ paramètres du vecteur complété pour chacun des f neurones de la couche F_2 . La complexité de l'algorithme serait donc de l'ordre de:

$$O(knf) \quad (2.7)$$

Puisque nous pouvons exprimer le nombre de neurones sur la couche F_2 du réseau de neurones comme le produit de la compression η et du nombre maximal de neurones possibles, soit le nombre de motifs dans la base d'entraînement n , nous obtenons :

$$O(kn^2/\eta) \quad (2.8)$$

Une fois parallélisé sur P nœuds de calculs, nous obtenons:

$$O(kn^2/\eta P) \quad (2.9)$$

Les ressources matérielles nécessaires à l'implémentation d'un réseau de neurones artificiels de type fuzzy ARTMAP sont utilisées pour stocker les données servant à l'apprentissage et au test ainsi que les neurones du réseau.

Si n est le nombre de patrons dans la base d'apprentissage, t le nombre de patrons dans la base de test et k le nombre de paramètres que contient chacun des patrons, il faudra donc $(n+t)(k+1)$ nombres réels pour stocker la base de données et $f(2k+1)$ nombres réels pour stocker le réseau.

2.3 Stratégies d'apprentissage pour les réseaux de neurones fuzzy ARTMAP

La stratégie d'apprentissage pour les réseaux de neurones fuzzy ARTMAP est une façon d'optimiser la performance du classificateur. Bien que l'utilisation des valeurs dites « standards » pour les hyperparamètres permet d'obtenir un bon taux de classification et de

compression, ces valeurs demeurent sous-optimales (Granger et al., 2007). Devant l'absence d'heuristique pour guider le choix des hyperparamètres et la complexité du problème d'optimisation non linéaire, l'utilisation d'algorithmes d'optimisation évolutive offre des résultats intéressants (Granger et al., 2007).

2.3.1 Optimisation par essaim de particules (OEP)

L'optimisation par essaim de particules (OEP) a été proposée par Kennedy et Eberhart (Kennedy et Eberhart, 1995). Étant basé sur le mouvement d'oiseaux en vol, l'OEP utilise le déplacement de particules dans un espace représentant les paramètres à optimiser.

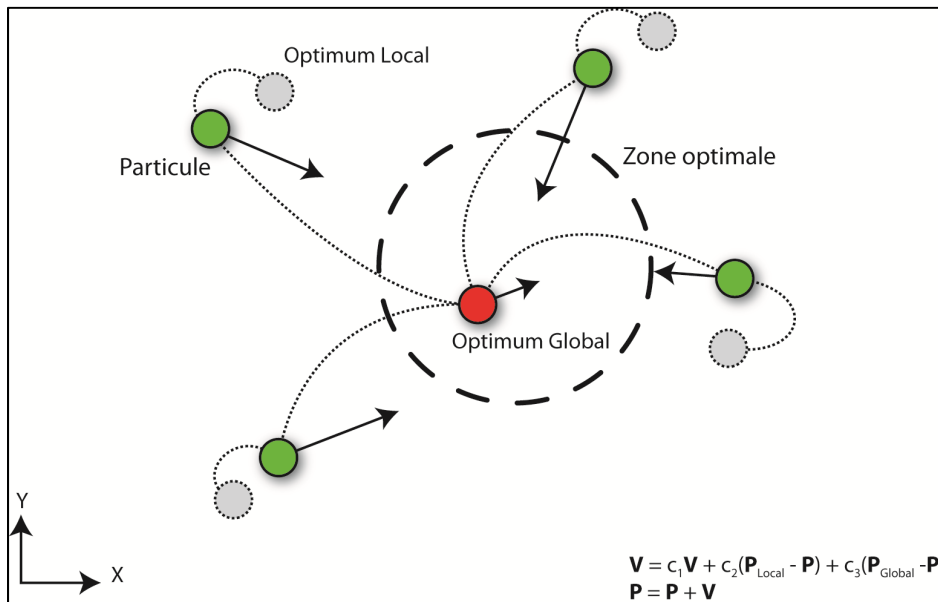


Figure 2.2 Optimisation par essaim de particules (OEP)

En conservant la position de leur meilleur résultat ainsi que la position du meilleur résultat obtenu dans tout le nuage de particules au fil des itérations, chacune des particules est capable de calculer un déplacement. Ainsi, une forme d'intelligence sociale est obtenue et évolue au fil du temps, et le nuage de particules va converger vers la zone optimale (Figure 2.2). L'Algorithme 2.3 décrit le fonctionnement de l'optimisation par essaims de particules.

Algorithme 2.3 La stratégie d'apprentissage OEP pour fuzzy ARTMAP

```

A. Initialisation:
    Fixer le nombre maximal d'itérations :  $q_{max}$ 
    Fixer les paramètres OEP :  $U, c_0, c_1, c_2, r_1$  and  $r_2$ 
    Fixer aléatoirement la position des particules tel que:  $\mathbf{p}_i = \mathbf{s}_i^0 \in [0;1]^d$ , pour  $i = 1$  à  $U$ 
    Fixer la vitesse des particules :  $\mathbf{v}_i^0 = 0$ , pour  $i = 1$  à  $U$ 
    Fixer la position de la meilleure performance globale  $\mathbf{p}_g = 0$ 
    Fixer le compteur d'itération  $q = 0$ 

B. Itérations:
    tantque  $q \leq q_{max}$  faire
        pour  $i = 1$  à  $U$  faire
            entraîner le réseau fuzzy ARTMAP avec  $\mathbf{s}_i^q$ 
            calculer la fonction objective  $F(\mathbf{s}_i^q)$ 
            si  $F(\mathbf{s}_i^q) > F(\mathbf{p}_i)$  alors
                Rafraichir la position de la meilleure performance locale:
                     $\mathbf{p}_i = \mathbf{s}_i^q$ 
            fin
            si  $F(\mathbf{s}_i^q) > F(\mathbf{p}_g)$  alors
                Rafraichir la position de la meilleure performance globale:
                     $\mathbf{p}_g = \mathbf{s}_i^q$ 
            fin
        fin
        pour  $i = 1$  à  $U$  faire
            Calculer la vitesse de la particule:
                 $\mathbf{v}_i^{q+1} = c_0^q \mathbf{v}_i^q + c_1 r_1 (\mathbf{p}_i - \mathbf{s}_i^q) + c_2 r_2 (\mathbf{p}_g - \mathbf{s}_i^q)$ 
            Calculer la position de la particule:
                 $\mathbf{s}_i^{q+1} = \mathbf{p}_i^q + \mathbf{v}_i^{q+1}$ 
            si  $\mathbf{s}_i^{q+1}$  dépasse ou est sur la limite de l'espace paramètres alors
                 $\mathbf{s}_i^q = \text{valeur limite}$ 
                 $\mathbf{v}_i^{q+1} = -\mathbf{v}_i^{q+1}$ 
            fin
        fin
         $q = q+1$ 
    fin

```

La technique d'apprentissage supervisé pour réseau de neurones fuzzy ARTMAP utilisant OEP (Granger et al., 2007) cherche à maximiser la fonction objective $F(\mathbf{s}_i^q)$ en utilisant les hyperparamètres de l'algorithme fuzzy ARTMAP pour définir l'espace paramètres dans lequel évoluent les particules de l'OEP. L'évaluation de la fonction objective consiste à tester le taux de classification d'un réseau de neurones avec les valeurs d'hyperparamètres donnés par la position de la particule OMOEP.

$$\mathbf{s}_q^i = (\alpha_q^i, \beta_q^i, \varepsilon_q^i, \rho_{0q}^i) \quad (2.10)$$

A la fin du processus d'optimisation, la capacité de la meilleure solution produite lors du processus à sa capacité de généraliser évaluée en mesurant le taux de classification obtenu à l'aide d'une dernière base de données de test qui n'a pas été utilisée durant l'apprentissage.

Puisque le temps nécessaire au calcul de la position et de la vitesse des particules est négligeable comparativement au temps de calcul de la fonction objective que la complexité du processus est donnée par le nombre de fois que la fonction objective est évaluée. Par conséquent, nous obtenons :

$$O(Uq_{max}kn^2/\eta P) \quad (2.11)$$

où U est le nombre de particules utilisé dans l'essaim et q_{max} le nombre maximal d'itérations dans l'optimisation. Ici aussi, il est possible de paralléliser le processus sur P processeurs, mais il est plus efficace de paralléliser au niveau particule qu'au niveau neurones dans ce type d'optimisation.

2.3.2 Optimisation multiobjective par essaim de particules (OMOEP)

L'optimisation multiobjective est la recherche d'un optimum lorsque la performance d'une solution dépend de plus d'un critère indépendant. Puisque ces aspects sont indépendants, il est impossible de les comparer sur un pied égal. La notion d'un ensemble de solutions non dominées est utilisée pour remplacer celle de la solution optimale. On dit que la solution A domine la solution B si les paramètres de la performance de la solution A sont supérieurs (ou inférieurs dans le cas d'une minimisation) à ceux de la solution B. Ainsi, une solution non dominée est une solution qui pour un des paramètres de performance est supérieure (ou inférieure) à toutes les autres solutions. L'ensemble des solutions non dominées, aussi appelé front de Pareto (Figure 2.3), représente l'enveloppe de performance du système.

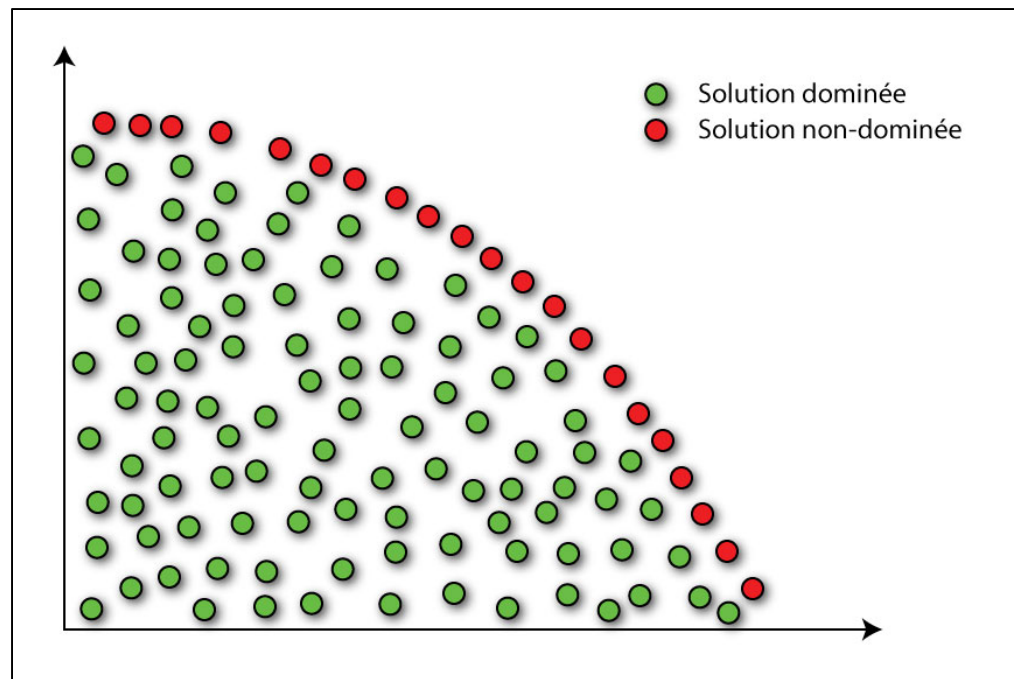


Figure 2.3 Un front de Pareto

L'algorithme d'optimisation multiobjective par essaim de particules (OMOEP) (Coello, Pulido et Lechuga, 2004) est une extension de l'algorithme OEP afin d'être capable de traiter les problèmes multiobjectifs. Tout comme dans le cas de l'OEP, OMOEP utilise une population de solutions qui évoluent dans le temps pour explorer l'espace paramètres en quête de la performance optimale. Par contre, la notion de la meilleure performance de toute la population (meilleure performance globale) a été remplacée par une archive de solutions non dominées produites au cours de l'évolution de la population. La meilleure performance d'une des particules (meilleure performance locale), a été remplacée par une archive locale pour contenir la position de l'histoire d'une particule dont la performance domine les autres.

Les changements apportés au concept d'optimum local et global de l'OEP pour étendre sa fonctionnalité aux problèmes multiobjectifs ont forcé un changement de la façon dont la vitesse des particules est calculée. Puisque l'optimum a été remplacé par un ensemble de solutions au lieu d'une seule comme dans le cas de l'OEP, une technique de sélection a été développée pour choisir une des solutions de l'archive à utiliser comme référence pour le calcul de vitesse. En divisant l'espace avec une hypergrille, on peut sélectionner un

hypercube contenant au moins une solution du front de Pareto, de façon à ce que la probabilité de choisir soit inversement proportionnelle au nombre de solutions que l'hypercube contient.

La tendance de l'OEP à converger rapidement sur un optimum local a aussi été atténuée par l'ajout d'un opérateur de mutation des solutions. Celui-ci permet de changer aléatoirement la position des particules pour assurer une bonne exploration. La probabilité qu'une solution subisse une mutation diminue de façon inversement proportionnelle au nombre d'itérations passées. La complexité de l'algorithme OMOEP est la même que celle de l'algorithme OEP puisque comme dans celui-ci, la complexité de l'évaluation domine grandement celle de l'évaluation de la position et de la vitesse de la particule.

Algorithme 2.4 La stratégie d'apprentissage OMOEP pour fuzzy ARTMAP

A. Initialisation:

Fixer le nombre maximal d'itérations : q_{max}

Fixer les paramètres OMOEP : U, w_0, c_1, c_2, r_1 et r_2

Fixer aléatoirement la position des particules tel que: $p_i = s_i^0 \in [0;1]^d$, pour $i = 1$ à U

Fixer la vitesse des particules : $v_i^0 = 0$, pour $i = 1$ à U

Fixer le compteur d'itération $q = 0$ et le compteur de solution dans l'archive $g = 0$

B. Itérations:

tantque $q \leq q_{max}$ **do**

pour $i = 1$ à U **faire**

 entraîner le réseau fuzzy ARTMAP avec s_i^q

 calculer la fonction objective $F(s_i^q)$

si $F(s_i^q)$ domine $F(p_i)$ **alors**

 Rafraichir la position de la meilleure performance: $p_i = s_i^q$

fin

pour $j = 1$ à g **faire**

if $F(s_i^q)$ domine $F(archive_j)$ **alors** $archive_j = \emptyset$

fin

fin

si $F(s_i^q)$ n'est pas dominé par une particule de l'archive **alors**

$archive = archive \cup s_i^q$

$g = g + 1$

fin

fin

pour $i = 1$ à U **faire**

 Sélectionner un hypercube non vide aléatoirement de façon à ce que la probabilité de son choix soit inversement proportionnelle au nombre de particules qu'il contient

 Choisir la particule (q_i) aléatoirement dans l'hypercube choisi

 Calculer la vitesse de la particule:

$$v_i^{q+1} = c_0^q v_i^q + c_1 r_1 (p_i - s_i^q) + c_2 r_2 (q_i - s_i^q)$$

 Calculer la position de la particule:

$$s_i^{q+1} = s_i^q + v_i^{q+1}$$

si s_i^{q+1} dépasse ou est sur la limite de l'espace paramètres **alors**

$s_i^q = \text{valeur limite}$

$$v_i^l = -v_i^q$$

fin

si s_i^q a une mutation **alors**

 Fixer s_i^q aléatoirement tel que $s_i^q \in [0;1]^d$

fin

fin

$q = q + 1$

fin

La technique d'apprentissage supervisé pour réseau de neurones fuzzy ARTMAP utilisant OMOEP (Granger, Prieur et Connolly, 2010), décrite dans l'Algorithme 2.4, cherche à maximiser la fonction multiobjective $F(s_q^i)$ en utilisant les hyperparamètres de l'algorithme fuzzy ARTMAP pour définir l'espace paramètres dans lequel évoluent les particules d'OMOEP. L'évaluation de la fonction multiobjective consiste à tester le taux de classification et la compression d'un réseau de neurones avec les valeurs d'hyperparamètres données par la position de la particule OMOEP, $s_q^i = (\alpha_q^i, \beta_q^i, \varepsilon_q^i, \rho_{0q}^i)$. La stratégie d'apprentissage supervisé pour fuzzy ARTMAP avec OMOEP permet d'optimiser à la fois le choix des hyperparamètres, la configuration du réseau ainsi que la valeur des poids synaptiques simultanément.

À la fin du processus d'optimisation, la capacité des solutions non dominées contenues dans l'archive ont leur capacité à généraliser évaluée en mesurant le taux de classification obtenu à l'aide d'une dernière base de données de test qui n'a pas été utilisée durant l'apprentissage.

2.4 Conclusion

Ce chapitre a tenté d'expliquer le fonctionnement de l'ACP, une technique de projection des données dans un sous-espace mathématique plus représentatif, lorsqu'elle est utilisée pour extraire et sélectionner les caractéristiques des motifs d'entrée. Elle permet aussi la réduction dimensionnelle des motifs d'entrée. Les réseaux de neurones fuzzy ARTMAP sont un type de classificateur neuronique capable de prédire l'identité d'un individu dans une RI en comparant le visage aux modèles faciaux créés durant le processus d'inscription. Ce type de réseau de neurones est capable d'un apprentissage stable et rapide et offre une bonne précision en généralisation. Son architecture et son fonctionnement le rendent léger, rapide et capable d'être parallélisé.

Il est possible d'utiliser un algorithme d'OEP, une technique d'optimisation exploitant l'intelligence sociale, pour sélectionner la valeur des hyperparamètres guidant le fonctionnement des réseaux de neurones fuzzy ARTMAP de façon à maximiser le taux de

classification positive. Par contre, ce processus dégrade la compression des réseaux entraînés. L'OMOEP étend la capacité de l'OEP aux problèmes multiobjectifs en trouvant un ensemble de solutions non dominées. L'utilisation de l'OMOEP permet de maximiser la performance des réseaux de neurones de type fuzzy ARTMAP en trouvant le meilleur compromis entre le taux de classification et la compression.

Dans le prochain chapitre, le processus d'implémentation sera discuté plus en détail. La sélection des processeurs ainsi que les modifications algorithmiques nécessaires à l'implémentation y seront abordées.

CHAPITRE 3

IMPLÉMENTATION MATÉRIELLE

Comme c'est souvent le cas, le passage du monde théorique de l'algorithme au monde réel du processeur, lors de l'implémentation d'un système, n'est pas aussi simple que cela puisse paraître. Le choix de l'algorithme et du support matériel pour l'exécuter doit être fait de façon informée et les deux doivent être ajustés pour obtenir une certaine synergie entre ces deux éléments. On ne peut se contenter de simplement choisir le processeur avec la plus haute cadence d'exécution ou le plus grand nombre de nœuds de calcul pour obtenir une meilleure performance. L'explosion des systèmes mobiles et distribués a aussi grandement modifié les contraintes d'implémentation en mettant l'accent sur la consommation d'énergie du système. Celle-ci se transpose dans la durée utile d'un appareil alimentée par une batterie rechargeable, qui est une caractéristique très importante pour les utilisateurs.

3.1 Sélection des processeurs

La gamme des technologies de nos jours disponibles pour implémenter un système dédié à la reconnaissance de visages dans des séquences vidéo utilisant les réseaux fuzzy ARTMAP est vaste et variée. Avec la vitesse à laquelle les différentes technologies et les outils de conception qui leurs sont associés évoluent, il devient difficile de bien distinguer laquelle est la plus intéressante. Choisir une implémentation matérielle qui peut atteindre le niveau de performance requis, tout en maintenant la possibilité d'être modifiée pour éviter l'obsolescence sans non plus négliger les ressources nécessaires au développement et support ni le cout par unité, est quelque chose de très complexe. Par exemple, le nombre limité de systèmes hautement spécialisés pour la biométrie dans le secteur public peut augmenter le coût par unité à un niveau élevé, voire même prohibitif, lorsque les ressources nécessaires au développement et soutiens sont considérés.

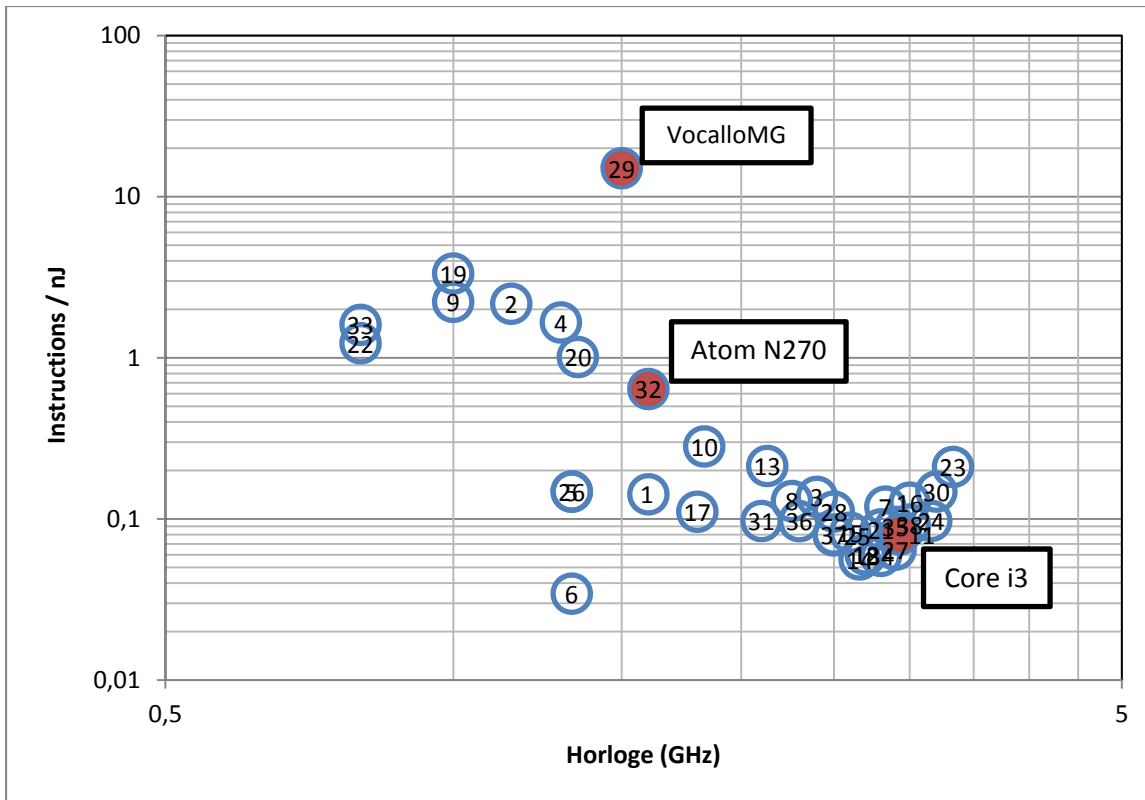


Figure 3.1 La fréquence d'horloge et la consommation d'énergie par instruction pour les processeurs sélectionnés

Trois processeurs commercialement disponibles ont été choisis afin de représenter différentes utilisations typiques. Ces processeurs sont : 1) l'Atom™ N270 de la compagnie Intel (Intel, 2009a), un processeur à simple cœur, basé sur une architecture simplifiée et une cadence d'horloge relativement basse afin d'obtenir une bonne consommation d'énergie, et qui est utilisé dans des applications mobiles, 2) le Core™ i3-530, aussi de la compagnie Intel (Intel, 2009b), un processeur à double cœur utilisant l'architecture de pointe d'Intel et visant les stations de travail économique comme principal segment de marché, et 3) le processeur Vocallo™ MGW de la compagnie Octasic (Octasic, 2004), avec quinze (15) cœurs indépendants et utilisant une technologie asynchrone pour offrir une très basse consommation d'énergie, et qui vise principalement le marché des systèmes distribués pour la conversion de signaux audio et vidéo.

La Figure 3.1 montre la cadence d'horloge (en gigahertz) ainsi que la consommation énergétique (en nanojoules) des trois processeurs retenus en rapport à un échantillonnage de différents autres processeurs commercialement disponibles. Les données rendues publiques dans les spécifications techniques par leurs fabricants concernant la performance des différents processeurs se trouvent dans le Tableau 3.1.

Le reste de cette section présente de façon plus détaillée les trois processeurs retenus pour l'implémentation.

Tableau 3.1 Caractéristiques des processeurs commercialement disponibles

#	Fabricant	Série	Modèle	Horloge (GHz)	Puissance (W)	Nœuds de calculs
1	AMD	Athlon	LE-1640	2.7	45	1
2	AMD	Athlon II x2	250	3	65	2
3	AMD	Athlon II x3	710	2.6	95	3
4	AMD	Athlon II x4	640	3	95	4
5	AMD	Athlon x2	7850	2.8	95	2
6	AMD	Athlon x2	5600	2.9	65	2
7	AMD	Athlon x2	4850e	2.5	45	2
8	AMD	Phenom x3	8850	2.5	95	3
9	AMD	Phenom x3	8450e	2.1	65	3
10	AMD	Phenom x4	9650	2.3	95	4
11	AMD	Phenom x4	9850	2.5	125	4
12	AMD	Phenom x4	9150e	1.8	65	4
13	AMD	Sempron	150	2.9	45	1
14	AMD	Sempron	140	2.7	45	1
15	ARM	Cortex	A8	1	0.3	1
16	ARM	Cortex	A9	1	0.45	1
17	Intel	Atom	N270	1.6	2.5	1
18	Intel	Atom	Z500	0.8	0.65	1
19	Intel	Atom	N470	1.83	6.5	1
20	Intel	Core 2	E8500	3.16	65	2
21	Intel	Core 2	Q9550	2.83	95	4
22	Intel	i3	i3-350M	2.26	35	2
23	Intel	i3	i3-530	2.93	73	2
24	Intel	i5	i5-520M	2.4	35	2
25	Intel	i5	i5-560UM	1.33	18	2
26	Intel	i5	i5-750	2.66	95	2
27	Intel	i7	i7-720QM	1.6	45	4
28	Intel	i7	i7-660UM	1.33	18	2
29	Intel	i7	i7-970	3.2	130	6
30	Intel	i7	i7-930	2.8	130	4
31	Intel	Itanium	9320	1.33	155	4
32	Intel	Xeon	L5630	2.13	40	4
33	Intel	Xeon	X5670	3.33	95	6
34	NVIDIA	Tesla	C2050	1.15	238	448
35	NVIDIA	Tesla	C1060	1.296	187.8	240
36	NVIDIA	Tesla	C870	1.35	170.9	128
37	Octasic	Vocallo	MGW	1.5	1.5	15
38	Xscale	StrongARM		0.8	0.5	1

3.1.1 Intel Core™ i3-530

Le processeur Core™ i3-530 de la compagnie Intel (Intel, 2009b) représente le type de processeur ordinairement trouvé dans les postes informatiques de bureau ainsi que les ordinateurs personnels. Il exécute le jeu d'instructions standard IA-32 de la compagnie Intel et il est supporté par la majorité des systèmes d'opérations modernes et des outils de développement commerciaux ainsi qu'en code source libre.

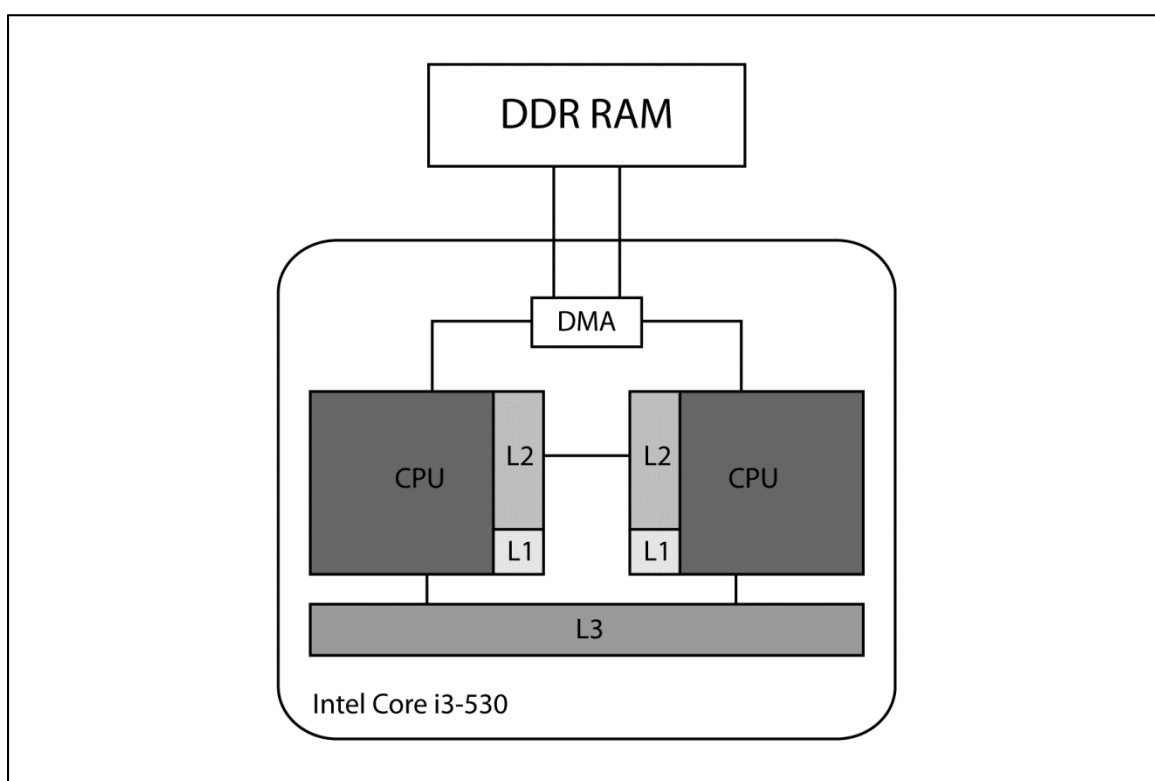


Figure 3.2 La configuration de la mémoire pour le processeur Intel Core i3-530

Ce processeur est disponible sur le marché depuis le premier trimestre de l'année 2010 et fait partie de la gamme de base de la famille de processeurs Core™. Basée sur la microarchitecture 32 nm Nehalem (Intel, 2008), ce processeur possède un contrôleur de mémoire DDR3 à deux canaux, une interface PCI Express, une interface Direct Media ainsi que les technologies Hyper-Threading et QuickPath.

Il possède deux nœuds de calcul évoluant à une cadence d'horloge de 2.93GHz pour puissance théorique de 73W. Le processeur possède 32kB de mémoire cache de niveau 1 par nœud de calcul (Figure 3.2), 256kB de mémoire cache de niveau 2 par nœud de calcul ainsi que 4 MB de mémoire cache de niveau 3 partagée par les deux nœuds de calcul (Intel, 2009b).

3.1.2 Intel Atom™ N270

Le processeur 32 bits Atom™ N270 de la compagnie Intel (Intel, 2009a) représente une alternative basse puissance standard. Ce type de processeur est utilisé dans des applications de type miniportatif ou des ordinateurs de petite taille. Il exécute le jeu d'instructions standard IA-32 de la compagnie Intel et il est supporté par la majorité des systèmes d'opération moderne et des outils de développement commerciaux ainsi qu'en code source libre. Il offre une alternative économique et basse puissance aux processeurs trouvés dans les stations de travail tout en restant compatible avec la plupart des programmes écrits pour ces derniers.

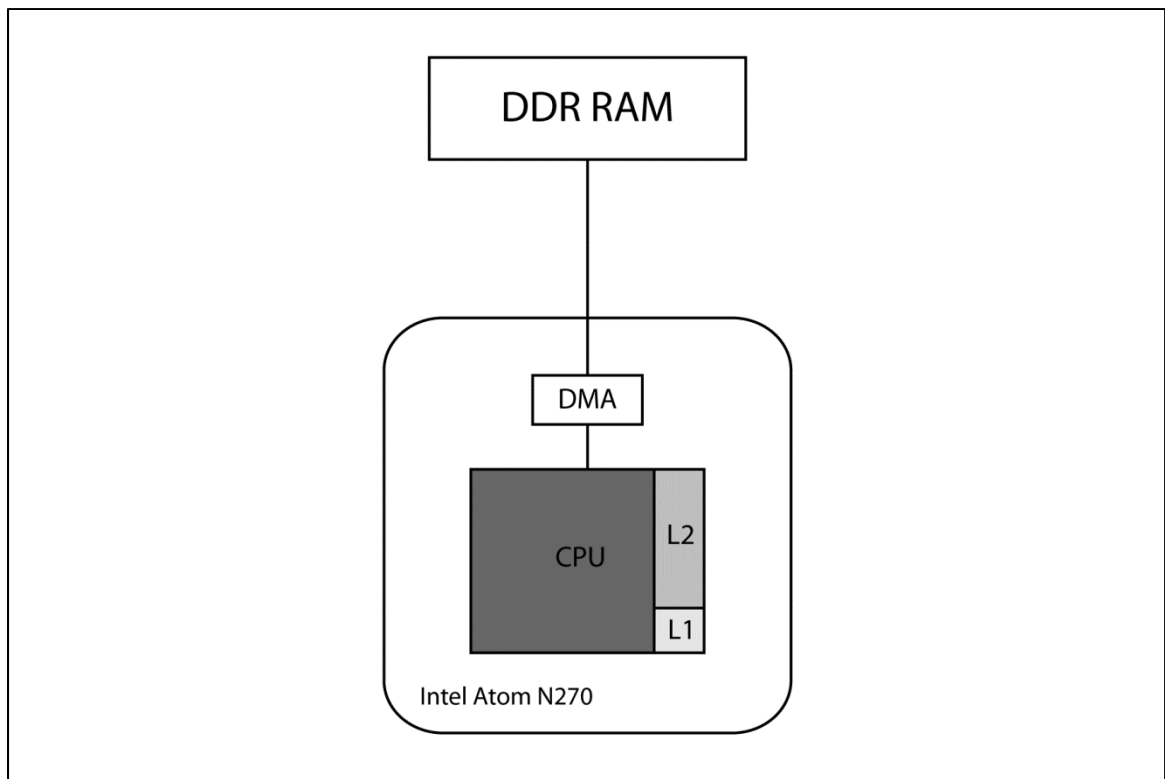


Figure 3.3 La configuration de la mémoire pour le processeur Intel Atom N270

Le processeur combine un seul nœud de calcul avec un contrôleur de mémoire (Figure 3.3) et de graphisme à l'intérieur du même circuit intégré de 45nm. Ceci permet au processeur d'évoluer à une cadence d'horloge de 1.6GHz pour une puissance théorique de 2.5W et une mémoire cache de niveau deux de 512kB (Intel, 2009a).

3.1.3 Octasic Vocallo MGW

Le processeur Vocallo MGW de la compagnie Octasic est l'alternative haute technologie. Ce processeur parallèle de très faible puissance vise principalement les applications de traitement de signaux numériques pour le transcodage audio & vidéo sur des réseaux IP (Octasic, 2004). Pour obtenir un bon degré de performance tout en maintenant une très faible consommation d'énergie, l'architecture Opus utilisée dans le processeur fonctionne de façon asynchrone permettant ainsi de réduire la consommation d'énergie en éliminant l'horloge globale (Tremblay, 2009).

Le Vocallo MGW Evaluation est une carte PCI de développement fabriquée par la compagnie Octasic, qui est composée d'un processeur Vocallo MGW, d'un contrôleur de mémoire DDR ainsi qu'une interface de télécommunication de type t041 (Octasic, 2004).

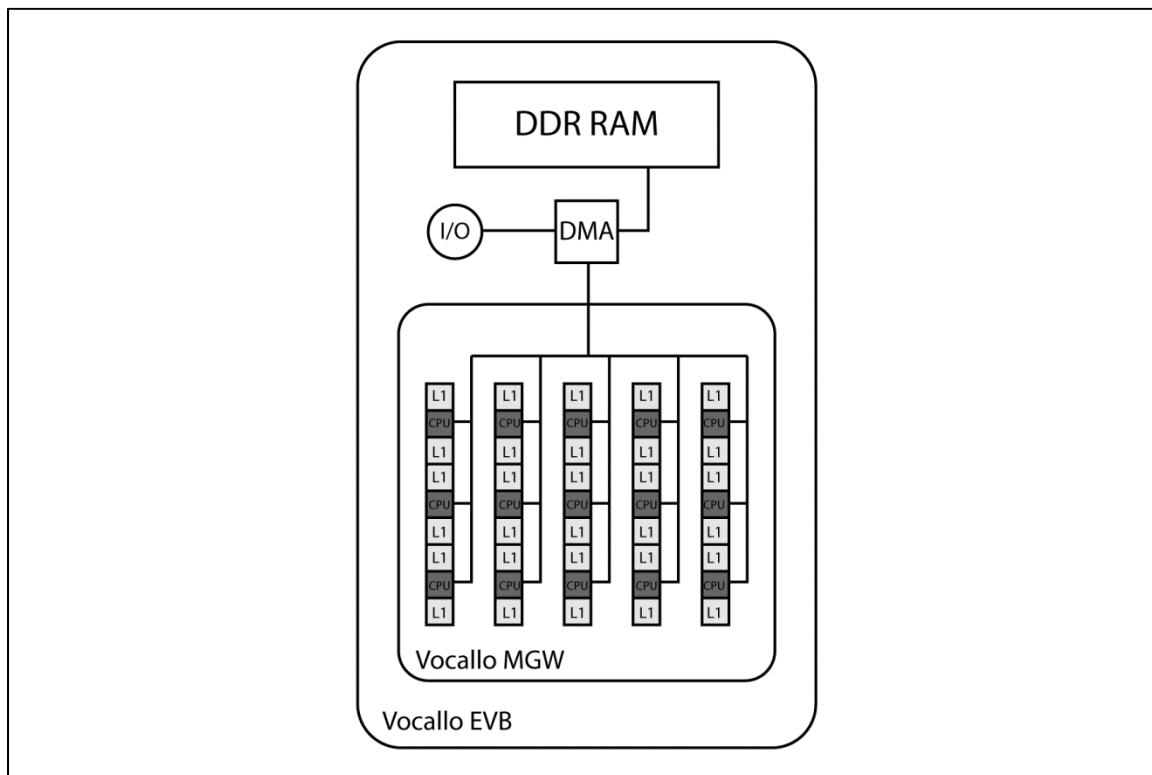


Figure 3.4 La configuration de la mémoire pour le processeur Vocallo MGW

Le processeur Vocallo MGW comporte 15 nœuds de calcul indépendants arrangés sous la forme d'une grille de trios par cinq (Figure 3.4). Chacun des nœuds de calcul est cerné par deux blocs de 48kB de mémoire cache niveau 1 pour un total de 96kB par nœud de calcul. Les nœuds de calcul peuvent lire et écrire dans la mémoire cache des autres nœuds ainsi qu'avec la mémoire RAM sur la carte de développement en utilisant un bus unique de 32bits opérant à une cadence de 1.5 GHz reliant tous les nœuds en parallèle et le contrôleur DMA.

Chacun des noeuds de calcul possède 16 unités de traitement arithmétique (UTA) fonctionnant en parallèle ainsi qu'une série de registres 32 bits. Les UTA utilisent un jeu

d'instruction de type RISC complémenté de fonctions spécifiques pour le traitement de signal numérique (Tremblay, 2009).

3.2 Modifications algorithmiques

En plus du choix du matériel sur lequel l'algorithme s'exécutera, l'algorithme fuzzy ARTMAP doit être adapté afin d'exploiter le matériel à son maximum. Un nombre minimal de modifications ont été apportées dans le but de bien pouvoir comparer les implémentations avec les systèmes actuels. Cependant, certains changements ont été effectués dans le but d'augmenter la performance et la portabilité. Les principales modifications sont listées ci-dessous.

3.2.1 Fonction de choix

La division de nombre réel est une tâche complexe et coûteuse en temps de calcul pour les processeurs modernes. Le problème est tel que certains processeurs plus spécialisés ne l'implémentent pas dans leur jeu d'instructions. Puisque la fonction de choix du réseau de neurones utilise une division de nombre réel et que cette opération est répétée un très grand nombre de fois, l'utilisation d'une fonction de choix de type « choix par ratio » (Carpenter et Gjaja, 1993), qui n'utilise pas la division de nombre réel, permet d'améliorer à la fois le temps d'exécution et la compatibilité avec des processeurs ne possédant pas cette opération dans leur jeu d'instructions. Ce choix par ratio est défini selon l'équation suivante :

$$T_j(A) = |A \wedge w_j| + (1 - \alpha)(2c - |w_j|) \quad (3.1)$$

3.2.2 L'inversion de la modification du signal de suivis

L'inversion de la modification du signal de suivi (Match Tracking) consiste à donner une valeur négative à l'hyperparamètre de suivi (ϵ). Cette modification à l'algorithme original permet d'améliorer la performance du classificateur lors de l'apprentissage de classes ayant des distributions se superposant (Carpenter, Milenova et Noeske, 1998).

En laissant la possibilité d'utiliser une valeur positive ou négative pour l'hyperparamètre de suivis ($\varepsilon \in [-1, 1]$), le choix d'utiliser ou non l'inversion de la modification du signal de suivi est laissé à la stratégie d'entraînement utilisant OMOEP.

3.2.3 Encodage binaire de la couche F_{AB}

Dans le but de minimiser l'espace mémoire requis pour entreposer le réseau de neurones, la couche F_{AB} du réseau est utilisée afin d'encoder en binaire le numéro associé avec la classe de l'individu au lieu d'avoir un neurone différent pour représenter chacune des classes. De cette façon, le nombre de classes L est réduit à $\log_2(L)$ neurones.

Puisque cet encodage est géré au niveau matériel, cette approche a aussi l'avantage de simplifier la tâche d'identifier la classe de l'individu en remplaçant une série de comparaisons par une simple lecture du nombre binaire donné par la couche F_{AB} .

3.2.4 Traitement parallèle

Avec l'augmentation de la consommation d'énergie qui est associée à la continuelle miniaturisation des transistors, l'astuce la plus commune pour augmenter la puissance de calcul tout en gardant le contrôle de la consommation d'énergie est d'ajouter des nœuds de calculs en parallèle. Par contre, le code machine provenant d'un algorithme classique ou sériel doit être modifié de façon à pouvoir être traité en parallèle et ainsi pouvoir tirer avantage de l'architecture parallèle du processeur.

Il existe deux grandes approches pour paralléliser l'algorithme fuzzy ARTMAP : le partitionnement des données et le partitionnement des neurones (Castro et al., 2004). Dans la première approche, les motifs d'entrée (dans ce cas-ci, les RI sous forme vectorielle) sont distribués parmi plusieurs instances du même réseau de neurones. Dans la deuxième approche, un gros réseau voit ses neurones divisés en plusieurs sous-réseaux qui vont calculer une prédiction locale à partir du même motif d'entrée avant de communiquer cette prédiction locale pour obtenir une prédiction globale à partir de celles-ci.

Puisque l'approche du partitionnement des données a montré une meilleure performance (Castro et al., 2004), la conception de l'implémentation est fortement guidée par la taille du réseau de neurones comparativement à la taille de la mémoire d'un des nœuds de calculs. Si la taille de la mémoire est suffisante pour garder le réseau entier, plusieurs instances du réseau peuvent être chargées dans la mémoire de chacun des nœuds de calcul et ainsi traiter plusieurs motifs d'entrée en parallèle. Bien que ceci est une bonne façon de gérer les cas où plusieurs visages se trouvent dans une même séquence ou bien si plusieurs caméras sont reliées à un seul classificateur, il n'y aura pas d'amélioration de la performance par rapport à l'utilisation d'un seul nœud de calculs s'il n'y a qu'une seule personne à identifier dans la séquence provenant d'une unique caméra. Dans ce cas, la vitesse d'exécution sera limitée par la vitesse d'acquisition de la caméra.

Dans le cas inverse, si la taille du réseau de neurones excède la taille de la mémoire locale d'un nœud de calcul, le réseau doit être divisé en sous-réseaux qui seront logés dans la mémoire locale des nœuds de calcul et serviront au calcul des prédictions locales avant de rassembler ces prédictions pour produire la prédiction globale.

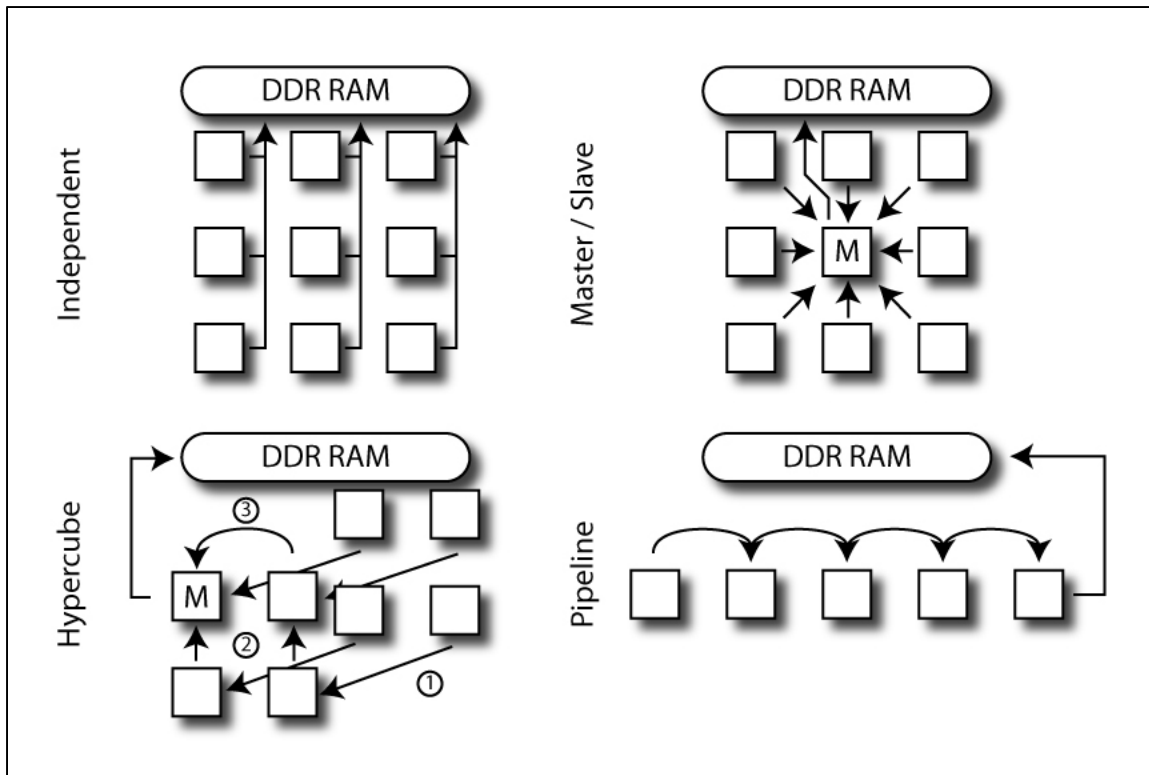


Figure 3.5 Les stratégies de communication

Les prédictions locales de chacun des nœuds de calcul peuvent être communiquées aux autres nœuds de plusieurs différentes façons (Figure 3.5). Des expériences ont montré le potentiel d'utiliser une communication de type hypercube (Malkani et Vassiliadis, 1995) ainsi qu'une communication de type pipeline (Castro et al., 2007). La communication de type indépendante représente le cas où il n'y a aucune communication entre les nœuds de calcul et la communication de type maître/esclave est le cas où chacun des nœuds esclaves communique avec un seul nœud maître qui recueille toutes les prédictions locales avant de produire la prédiction globale.

3.2.5 Prédiction par accumulation d'informations

Jusqu'à présent, les mentions du taux de classification faisaient référence au taux de classification pour une seule trame. Avec l'utilisation d'un module de trace, plusieurs trames

d'une séquence vidéo peuvent être combinées afin d'améliorer la qualité de la prédiction de l'identité de l'individu cible.

Une des façons les plus simples et les plus efficaces pour exploiter le traitement des données temporelles est de placer les prédictions faites pour chacune des trames dans un tampon de décision. Ceci permet de créer une estimation discrète de la distribution de la probabilité à partir de laquelle la prédiction finale est faite (Carpenter et Ross, 1995).

Ceci permet d'augmenter le taux de classification jusqu'au point désiré, mais, en contrepartie, crée un délai avant que la prédiction finale durant la période où le tampon de décision se remplit. Ceci peut causer un problème lorsque l'application nécessite un traitement en temps réel.

3.3 Conclusion

Dans ce chapitre les trois processeurs sélectionnés pour représenter le spectre des solutions matérielles disponibles sont l'Atom™ N270 de la compagnie Intel, le Core™ i3-530 aussi de la compagnie Intel et le processeur Vocallo™ MGW de la compagnie Octasic.

Le Core™ i3-530 est un processeur multicœur communément trouvé dans les ordinateurs de bureau. Il fonctionne avec une cadence d'horloge de 2.93 GHz et une puissance théorique de 73W. L'Atom™ N270 est un processeur simple cœur basse puissance trouvé dans des appareils électroniques mobiles. Il fonctionne à une cadence d'horloge de 1.6 GHz et une puissance théorique de 2.5W. Le Vocallo™ MGW est un processeur multicœur basse puissance pour le transcodage et les appareils mobiles et distribués. Il fonctionne à une cadence approximative de 1.5 GHz pour une puissance théorique de 1.5W.

La performance et la portabilité de l'algorithme fuzzy ARTMAP ont été améliorées en le modifiant pour utiliser une fonction de choix qui n'utilise pas l'opérateur de division, en inversant la modification du signal de suivi, en encodant de façon binaire la couche de

neurones F_{AB} , en parallélisant l'algorithme et en le modifiant pour améliorer la prédiction de l'identité à l'aide de l'accumulation d'information.

Dans le chapitre suivant, la méthodologie utilisée pour évaluer la performance des différentes implémentations matérielle sera présentée. La base de données, la méthodologie expérimentale ainsi que les indicateurs de performance y seront décrits.

CHAPITRE 4

MÉTHODOLOGIE D'ÉVALUATION

L'application présentée dans cet ouvrage essaie de reproduire un scénario d'identification biométrique par la reconnaissance du visage pour contrôler l'accès à une pièce d'équipement ou un espace sécurisé. Contrairement à un scénario de type « validation », l'utilisateur n'a pas besoin de s'identifier au système (ex. : nom d'utilisateur, numéro d'accès, etc.) qui vérifie si l'affirmation d'identité est valide ou non. Dans une application de type « identification », le signal biométrique est comparé à tous les individus inscrits dans le système pour prédire l'identité la plus probable de l'individu cible (Jain, Ross et Prabhakar, 2004).

4.1 Base de données expérimentale

La base de données IIT-NRC (Gorodnichy, 2006) a été retenue pour évaluer la performance des diverses implémentations du système. Cette base de données a été spécialement conçue pour l'étude du contrôle d'accès à l'aide de la reconnaissance du visage dans des séquences vidéo. Elle a été créée en utilisant du matériel commercialement disponible de type caméra intégré ou « webcam » qu'on peut trouver dans les ordinateurs personnels ainsi que les applications mobiles.

La base de données contient 11 paires de séquences (une séquence pour l'apprentissage et une séquence pour l'évaluation de la performance) provenant chacune d'un individu différent. Les séquences vidéo de type MPG1 ont une résolution de 120x160 pixels, une durée d'environ 15 secondes et ont été échantillonnées à une fréquence de 20 trames par seconde. La caméra a été orientée de façon à s'assurer que le visage de l'individu présent dans la séquence occupe entre 1/8 et 1/4 de l'image et les conditions d'échantillonnage ont été les mêmes pour les deux séquences. La détection de visage a été effectuée pour chacune des trames des séquences vidéo à l'aide du détecteur de visage de type Viola-Jones (Viola et Jones, 2004) offert dans la librairie OpenCV. Le nombre de RI détectés pour une séquence

varie entre 40 à 190 RI par personne. Ceci donne donc des taux de détection moyens de 47.3 % et variants entre 13.3 % et 63.3 % en fonction de l'individu. La base de données issue de la séquence vidéo utilisée pour l'apprentissage est composée de 1527 RI tandis que celle utilisée pour le test est composée de 1585 RI.

Les images ont été recadrées et ont eu leurs orientations corrigées de façon à aligner les yeux horizontalement avant de voir leurs résolutions diminuées pour produire des RI de 24x24 pixels. Cette taille de RI a été spécifiquement choisie pour conserver 12 pixels entre les yeux de l'individu cible. Cette basse résolution est connue pour être suffisante pour la reconnaissance de visages dans des séquences vidéo (Gorodnichy, 2006). Puisqu'il n'y a qu'un seul individu par séquence vidéo, nous pouvons nous permettre de supposer une performance idéale du module de suivi.

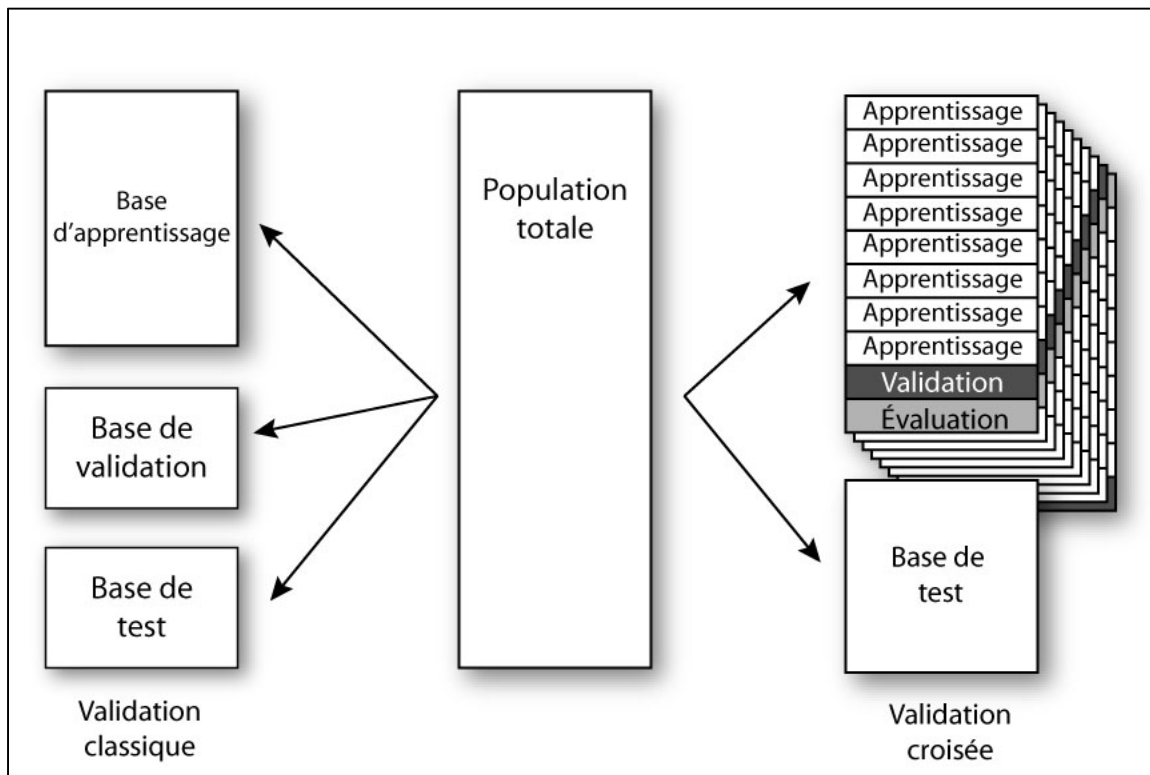


Figure 4.1 Validation croisée modifiée

La base de données d'entraînement a été subdivisée pour produire les différents blocs de données utilisées pour chacune des réplifications statistiques (Figure 4.1). Pour créer les blocs de données, une technique de validation croisée à k-niveaux (Duda, Hart et Stork, 2001) a été utilisée pour générer la base d'apprentissage, la base de validation et la base d'évaluation pour chacune des dix réplifications. Dix niveaux de réplification ont été jugés comme le compromis idéal entre le besoin de réplifications indépendantes et le peu de données disponibles pour l'entraînement.

4.2 Méthodologie expérimentale

Afin de réduire la dimensionnalité des RI avant de les utiliser comme motifs d'entrée pour le réseau de neurones, l'analyse par composantes principales (ACP) (Turk et Pentland, 1991), aussi connue comme la transformation Karhunen-Loeve, est utilisée pour extraire l'information d'un plus haut niveau contenue dans la matrice de luminosité des RI présentée sous forme vectorielle. La transformation ACP est calculée à l'aide de la base de données d'apprentissage et est par la suite appliquée à tous les blocs de données (apprentissage, validation, évaluation et test) utilisés pour la réplification en vigueur.

Un ensemble de 32 solutions non dominées ou moins est ensuite produit avec l'usage de la stratégie d'apprentissage OMOEP pour les réseaux de neurones de type fuzzy ARTMAP. L'algorithme d'optimisation est initialisé avec 32 particules et exécute 25 itérations. Ce processus est repris pour chacun des nombres de caractéristiques conservées différents et le tout est répété dix fois pour chacune des réplifications statistiques.

Les solutions ainsi produites ont été testées sur chacune des trois plateformes matérielles choisies afin d'obtenir les différentes mesures de performances de l'implémentation. Pour présenter les données sous forme de tableaux, trois solutions typiques ont été sélectionnées de l'ensemble produit pour représenter la variété de solutions produites :

- la solution « lourde » est le réseau de neurones avec le meilleur taux de classification et qui peut être contenu dans la mémoire locale du processeur choisi;
- la solution « moyenne » est le réseau de neurones avec le meilleur taux de classification capable d'être contenu dans la mémoire locale de trois nœuds de calcul du processeur Vocallo MGW;
- la solution « légère » est le réseau de neurones avec le meilleur taux de classification capable d'être contenu dans la mémoire local d'un seul nœud de calcul du processeur Vocallo MGW;
- la solution « complète » est le réseau de neurones avec le meilleur taux de classification sans utiliser l'analyse par composantes principales.

L'effet que produit la taille du tampon de décision utilisé pour l'accumulation de preuves a été mesuré en faisant la moyenne des taux de classification obtenus par toutes les permutations possibles dans les séquences en utilisant le fenêtrage. Ceci a été répété pour chaque taille de tampon de décision possible et chacune des séquences vidéo dans la base de données de test. La moyenne des résultats pour toutes les séquences a été calculée pour chaque taille de tampon de décision et est présentée comme la performance moyenne du réseau en fonction de la taille du tampon de décision.

Afin de pouvoir caractériser la communication inter-nœuds de calculs sur le processeur Vocallo MGW, les techniques de communication (Figure 3.5) maître/esclave, pipeline et indépendant ont été testées avec la solution « moyenne » utilisant 16 composantes ACP. Le temps passé à communiquer avec la mémoire RAM, le temps passé à communiquer avec les autres nœuds de calcul, le temps passé en attente de communication ainsi que le temps total de traitement ont tous été mesurés à l'aide de la minuterie incluse dans le processeur Vocallo MGW.

4.3 Indicateurs de performance

Pour comparer les différentes implémentations du système de reconnaissance de visages dans des séquences vidéo, quatre différents aspects de la performance du système ont été

quantifiés. Ils sont : le taux de classification, l'espace mémoire nécessaire, le temps de traitement ainsi que la consommation d'énergie.

Le taux de classification est le ratio de bonnes prédictions faites par rapport au nombre de prédictions totales faites par le système. Il décrit la probabilité du système à fournir une bonne prédiction de l'identité de l'individu cible.

L'espace mémoire nécessaire représente la taille qu'occupe le réseau de neurones dans la mémoire RAM du système. Il est fonction du nombre de caractéristiques conservées dans les motifs d'entrée ainsi que le nombre de neurones présents dans le réseau une fois entraîné.

Le temps de traitement est le temps moyen nécessaire afin de traiter une seule RI par le système. Il est obtenu en divisant le temps requis pour traiter la base de données entière et de diviser cette valeur par le nombre de RI présents dans la base de données.

La consommation d'énergie est estimée en divisant le nombre d'instructions exécutées durant le traitement de la base de données par le nombre d'instructions par joule associé au processeur utilisé. Le nombre d'instructions exécutées varie en fonction du nombre de neurones présents dans le réseau de neurones utilisés et le nombre de caractéristiques conservées dans les motifs d'entrée. Un exemple de calcul est présenté dans l'Annexe I

4.4 Conclusion

Il a été possible de voir dans ce chapitre que la base de données utilisée pour évaluer la performance des implémentations est celle de l'IIT-CNRC. Cette base de données contient des séquences vidéo typiques d'une application de contrôle d'accès. Elle contient deux séquences vidéo pour chacun des 11 individus compris dans la base de données. La détection de visages a été effectuée à l'aide des outils de la librairie OpenCV et a donné un taux de détection moyen de 47.3 %. Les RI créées sont de 24 x 24 pixels. Une validation croisée à

deux niveaux est utilisée pour pouvoir répéter le processus dix fois de façon à obtenir une mesure de performance moyenne avec peu de données disponibles.

La méthodologie expérimentale consiste à appliquer l'ACP au bloc de données, d'entraîner un ensemble de réseaux de neurones fuzzy ARTMAP à l'aide de la stratégie d'apprentissage OMOEP et évaluer la performance des solutions retenues sur les trois processeurs sélectionnés. Les indicateurs de performance utilisés sont le taux de classification positive, l'espace mémoire nécessaire, le temps de traitement ainsi que la consommation d'énergie.

Dans le prochain chapitre, les résultats expérimentaux seront présentés et analysés.

CHAPITRE 5

RÉSULTATS EXPÉRIMENTAUX

L'expérimentation faite pour l'implémentation de l'algorithme fuzzy ARTMAP pour la reconnaissance de visages dans des séquences vidéo est divisée en quatre parties : les techniques de communication internœuds, l'extraction et la sélection de composantes, la performance de l'implémentation matérielle et la taille du tampon de décision.

5.1 Communications internœuds

Dans le cas du processeur Atom N270 d'Intel, la question de la communication internœuds ne s'applique pas à cause de son architecture à simple nœud de calcul. Pour le Core i3 d'Intel, le traitement de la communication internœuds se fait partiellement au niveau matériel, en exploitant des technologies de partage de mémoire cache comme QuickPath, et partiellement au niveau de l'OS, avec l'utilisation de la librairie OpenMP pour le traitement parallèle. Mais pour le Vocallo MGW d'Octasic, la première question qui nécessitait une réponse était comment communiquer les prédictions locales entre les nœuds de calculs afin d'arriver à une prédiction globale.

La Figure 5.1 présente le temps total nécessaire au traitement de la base de test sur le processeur Vocallo MGW pour les trois différents types de communication : maitre/esclave, pipeline et indépendant. Le temps de traitement est la somme du temps de calcul, du temps passé à communiquer avec la mémoire RAM, du temps passé à communiquer entre nœuds de calculs ainsi que le temps passé en attente de communiquer.

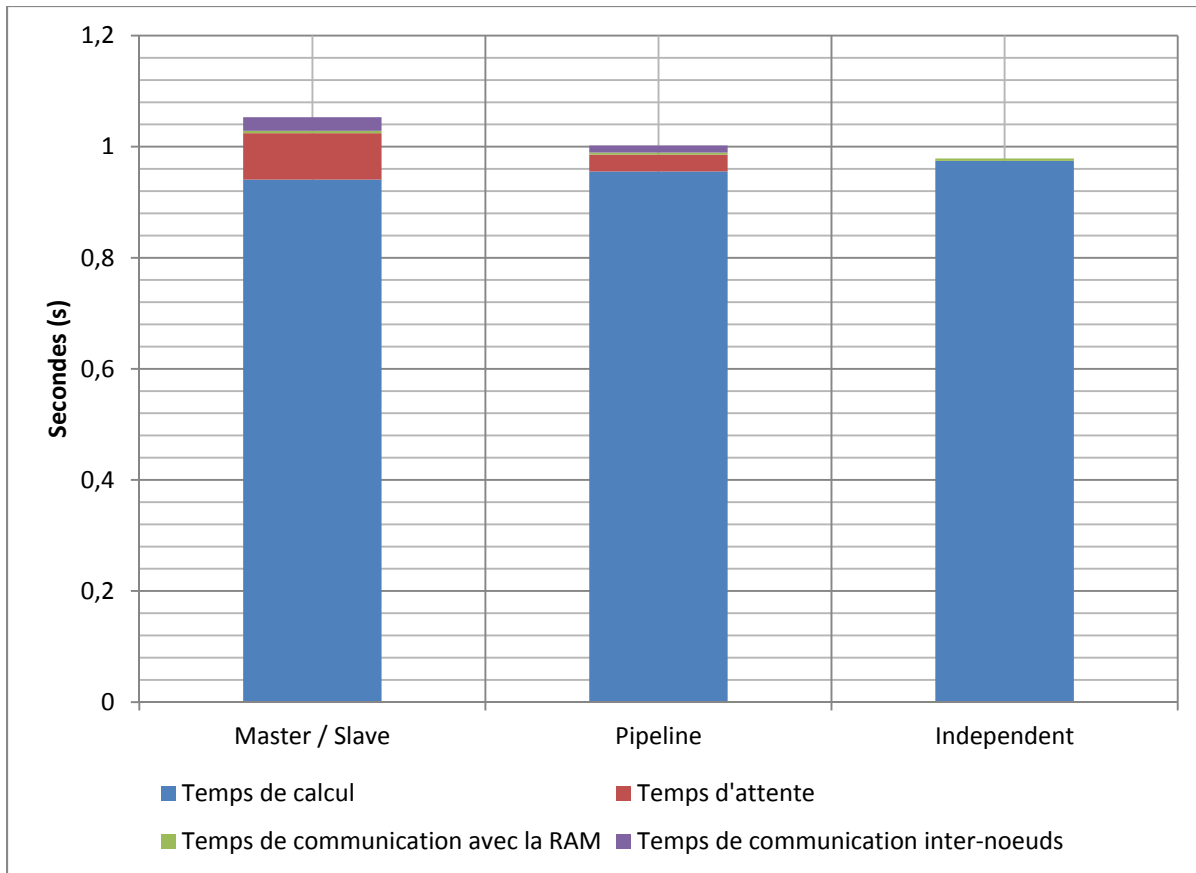


Figure 5.1 Temps de traitement et communication pour les différentes stratégies de communication

Une des pièces d'information la plus importante qui se trouve dans ces résultats est que le temps de calcul domine largement le temps passé à communiquer. Par conséquent, le choix de la méthode de communication n'a qu'une très faible influence sur le temps de traitement total. La Figure 5.1 montre aussi que le temps perdu à attendre pour communiquer est légèrement supérieur pour la méthode maître/esclave que pour la méthode pipeline. Ceci s'explique par le fait que le nœud maître ne peut communiquer qu'avec un seul nœud esclave à la fois.

Il y a une augmentation du temps passé à communiquer entre les nœuds et du temps passé à attendre pour pouvoir communiquer lorsque les neurones du réseau sont distribués sur tous les nœuds en comparaison avec lorsque les neurones du réseau satureront la mémoire cache du

plus petit nombre de nœuds possibles. Ceci est causé par le plus grand nombre de prédictions locales qui ont besoin d'être communiquées pour obtenir la prédiction finale. Puisque le temps de calcul est intimement lié au nombre de neurones présents dans le sous-réseau local, le temps de traitement lorsque les neurones sont distribués parmi tous les nœuds de calculs est inférieur au temps de traitement lorsque les neurones sont regroupés dans le plus petit nombre de nœuds possibles. La technique de communication indépendante n'ayant aucune communication inter-nœuds voit son temps uniquement divisé entre le temps de calcul et le temps de communication avec la mémoire RAM.

Conséquemment, le choix de la technique de communication utilisée avec le processeur Vocallo MGW a une influence négligeable sur le temps de traitement total. Le faible coût de la communication permet donc la flexibilité de choisir entre une solution plus rapide, avec plusieurs instances d'un réseau avec peu de neurones chargés dans les nœuds de calculs, ou bien une solution plus précise, avec un réseau ayant beaucoup de neurones répartis entre les multiples nœuds de calculs. De plus, même si l'on voit un avantage à distribuer les neurones sur le plus grand nombre de nœuds de calculs possible, le temps de traitement total reste supérieur au temps de traitement obtenu lorsque le réseau est concentré dans le plus petit nombre de nœuds de calculs possible pour pouvoir partager les RI entre plusieurs instances du réseau.

Puisque le choix de la technique de communication a peu d'influence sur le temps de traitement total, la technique maître/esclave a été utilisée pour le reste de l'expérimentation, car elle permet une meilleure précision que l'approche complètement indépendante et simplifie l'implémentation du tampon de décision.

5.2 Extraction et sélection de caractéristiques

Le choix du nombre de caractéristiques à garder après l'analyse par composantes principales est critique, car il influence tous les aspects de la performance de l'implémentation. Si trop peu de caractéristiques sont conservées, il manque d'informations pour bien différencier les

individus et si trop de caractéristiques sont conservées, de l'information non pertinente est conservée et cause de la confusion dans le processus d'identification.

Le nombre de caractéristiques retenues a aussi une influence sur la taille du motif d'entrée du classificateur et par conséquent, la quantité de mémoire nécessaire pour entreposer le réseau de neurones grandit exponentiellement avec le nombre de caractéristiques conservées. Ceci devient vite un problème puisque la taille du réseau de neurones influence aussi le temps de traitement ainsi que la consommation d'énergie.

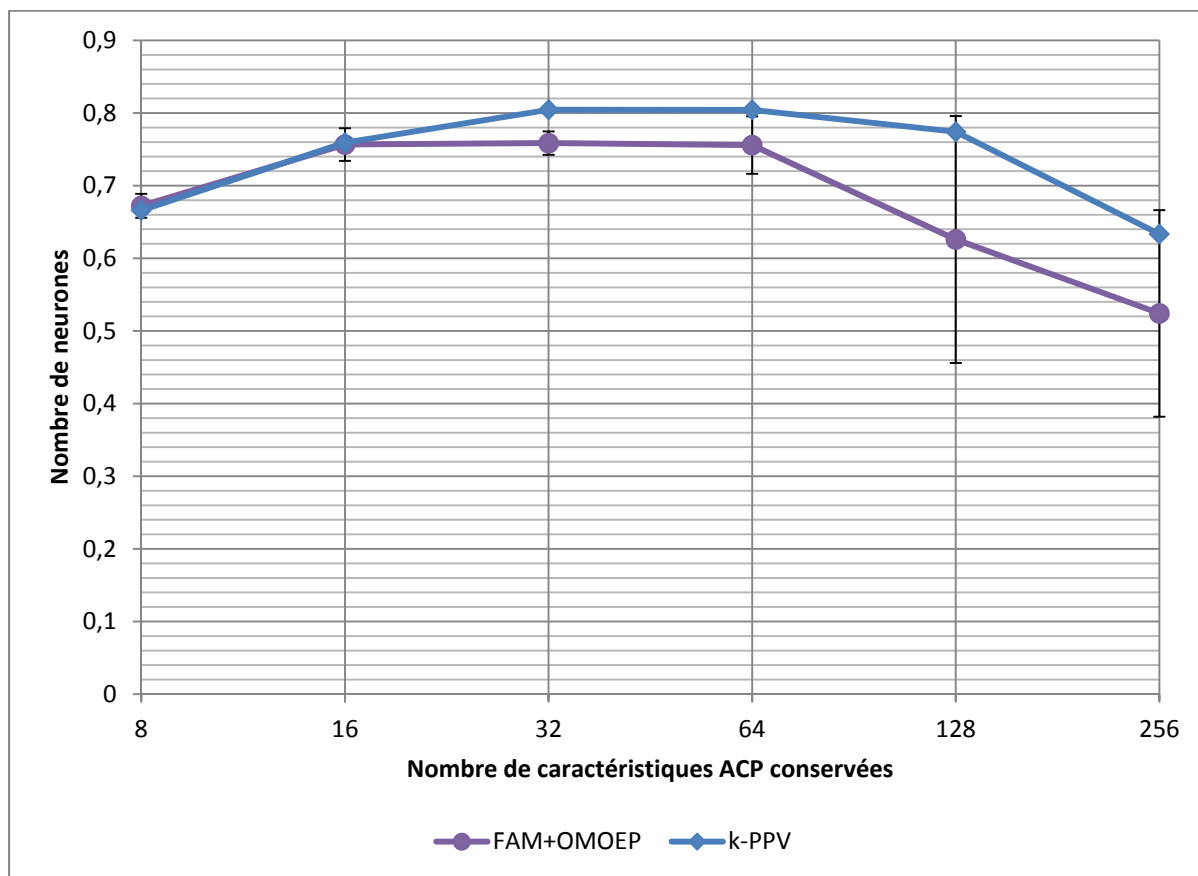


Figure 5.2 Le taux de classification en fonction du différent nombre de caractéristiques conservées

Bien que la meilleure façon d'explorer l'influence du nombre de caractéristiques retenues serait d'entraîner des réseaux de neurones en faisant varier le nombre de caractéristiques,

cette approche n'est pas pratique, car elle nécessiterait beaucoup de temps pour la réaliser. En effet, l'entraînement de réseaux de neurones en utilisant la stratégie OMOEP requiert plusieurs heures de calcul sur un serveur multiprocesseur moderne. Ainsi, il est très utile d'avoir une façon d'estimer l'influence du nombre de caractéristiques retenues sur le taux de classification du système sans avoir à entraîner tous les réseaux à l'aide de l'OMOEP. La Figure 5.2 montre le taux de classification obtenu par le réseau le plus précis de l'ensemble produit à l'aide d'OMOEP comparé au taux de classification obtenu par un classificateur de type k-Plus Proches Voisins (k-PPV), tous les deux étant fonction du nombre de caractéristiques retenues.

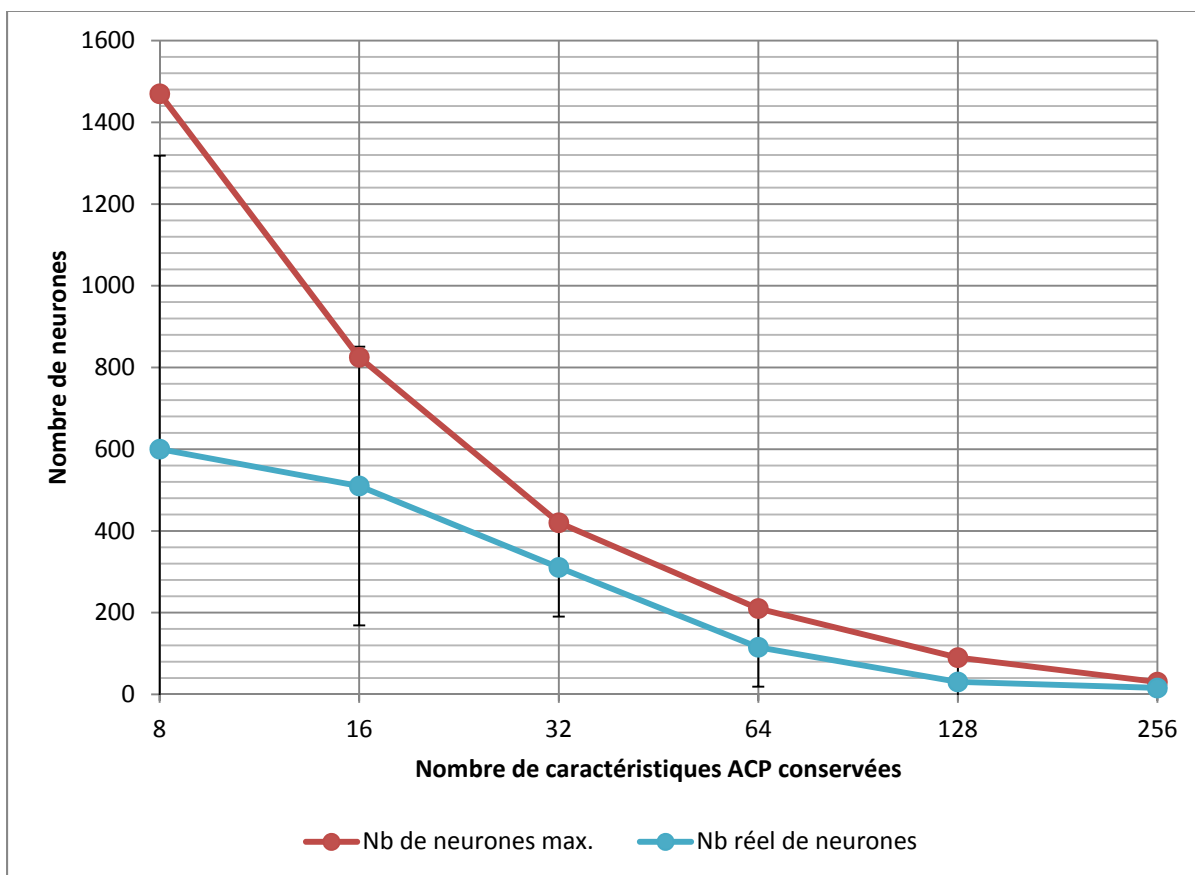


Figure 5.3 Le nombre de neurones dans le réseau entraîné ainsi que le nombre maximum de neurones que peut contenir la mémoire cache du Vocallo MGW en fonction du nombre de caractéristiques conservées

Même s'il y a un peu de divergence entre les résultats, spécialement dans le cas où un plus grand nombre de caractéristiques sont conservées, les taux de classification du réseau de neurones et du classificateur k-PPV suivent la même tendance.

La Figure 5.3 montre le nombre maximal de neurones que la mémoire cache d'un nœud de calcul du processeur Vocallo MGW en fonction du nombre de caractéristiques conservées. Puisque la taille du réseau augmente de façon exponentielle avec le nombre de caractéristiques conservées, il est préférable de choisir le plus petit nombre de caractéristiques possibles avant de voir le taux de classification diminuer de façon marquée. Dans le cas présent, le point se trouve près de 16 caractéristiques. Des réseaux utilisant 32 et 64 caractéristiques ont été entraînés pour illustrer l'effet de la taille du réseau sur le taux de classification, le temps de traitement ainsi que la consommation d'énergie.

5.3 La performance en reconnaissance d'images individuelles

Dans la Figure 5.4, les différents ensembles regroupant les solutions pour chacune des répliques statistiques des séquences d'optimisation, en fonction de différents nombres de caractéristiques conservées dans les motifs d'entrée, sont présentés. L'axe vertical de la figure représente le taux de classification d'une solution et l'axe horizontal sa valeur de compression. La valeur de la compression des solutions est mieux représentée à l'aide d'une échelle logarithmique puisqu'elle révèle les fronts de Pareto continus produits par l'algorithme d'optimisation multicritères.

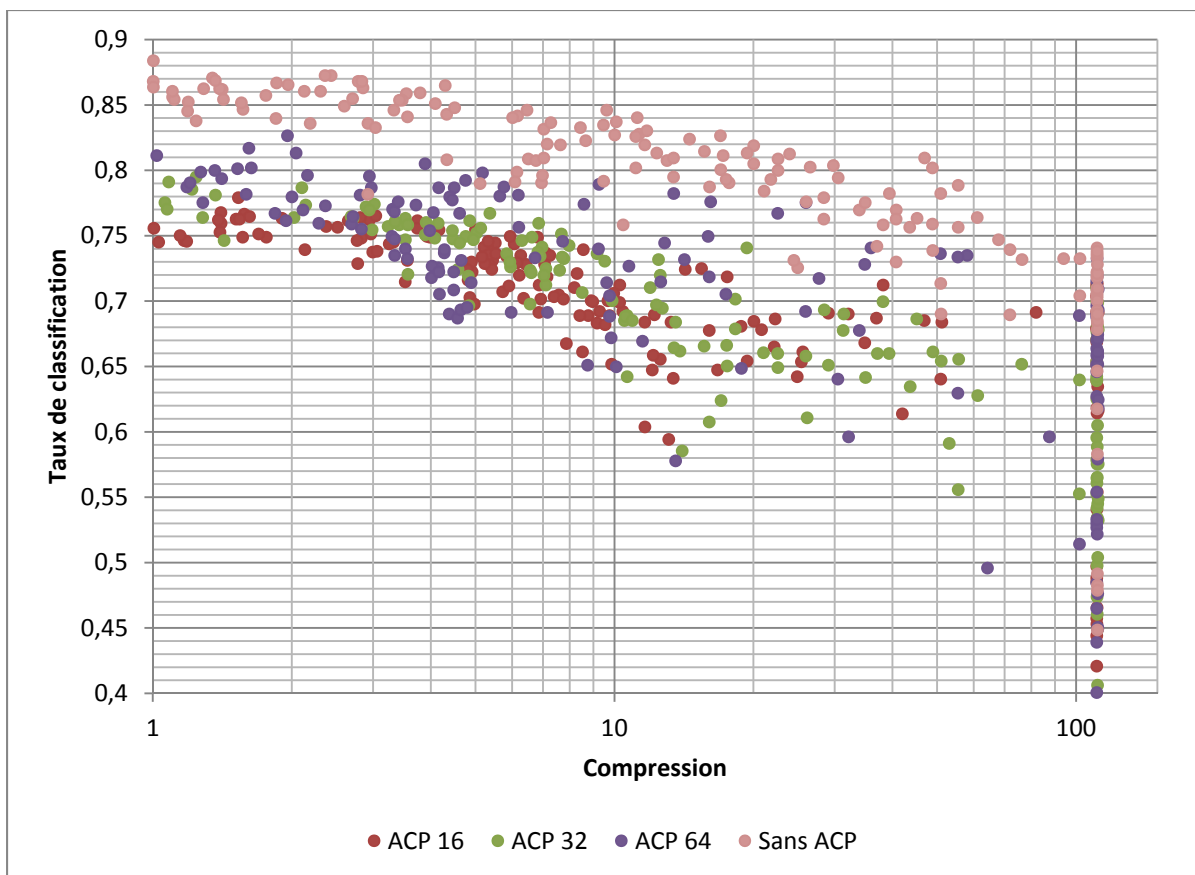


Figure 5.4 Fronts Pareto pour les différentes répliques statistiques en fonction du nombre de caractéristiques conservées dans le motif d'entrée

Une des premières pièces d'information visibles dans les résultats de la Figure 5.4 est l'effet du nombre de caractéristiques conservées sur le taux de classification des solutions. Il y a une nette différence entre le taux de classification obtenu à l'aide de motifs utilisant toute l'information provenant des RI et celui obtenu à l'aide de motifs provenant de RI transformés avec l'ACP pour réduire leur dimensionnalité. Il est aussi possible de voir que le taux de classification des solutions à faible compression augmente avec le nombre de composantes conservées. Considérant la grande variance du taux de classification des solutions possédant une forte compression, la différence est moins flagrante lorsque le taux de compression augmente.

Puisque la compression n'est fonction que du nombre de neurones que contient le réseau et ne tient pas compte de la taille du motif d'entrée, il est difficile de bien comparer les différentes solutions ainsi produites. En transformant la valeur de compression en l'espace actuel que le réseau occupe en mémoire, tel que présenté dans la Figure 5.5, les solutions non dominées utilisant différents nombres de caractéristiques dans leurs motifs d'entrée peuvent être comparées sur un même pied. Ceci permet d'identifier facilement la solution produite qui est le plus proche du point d'opération désiré.

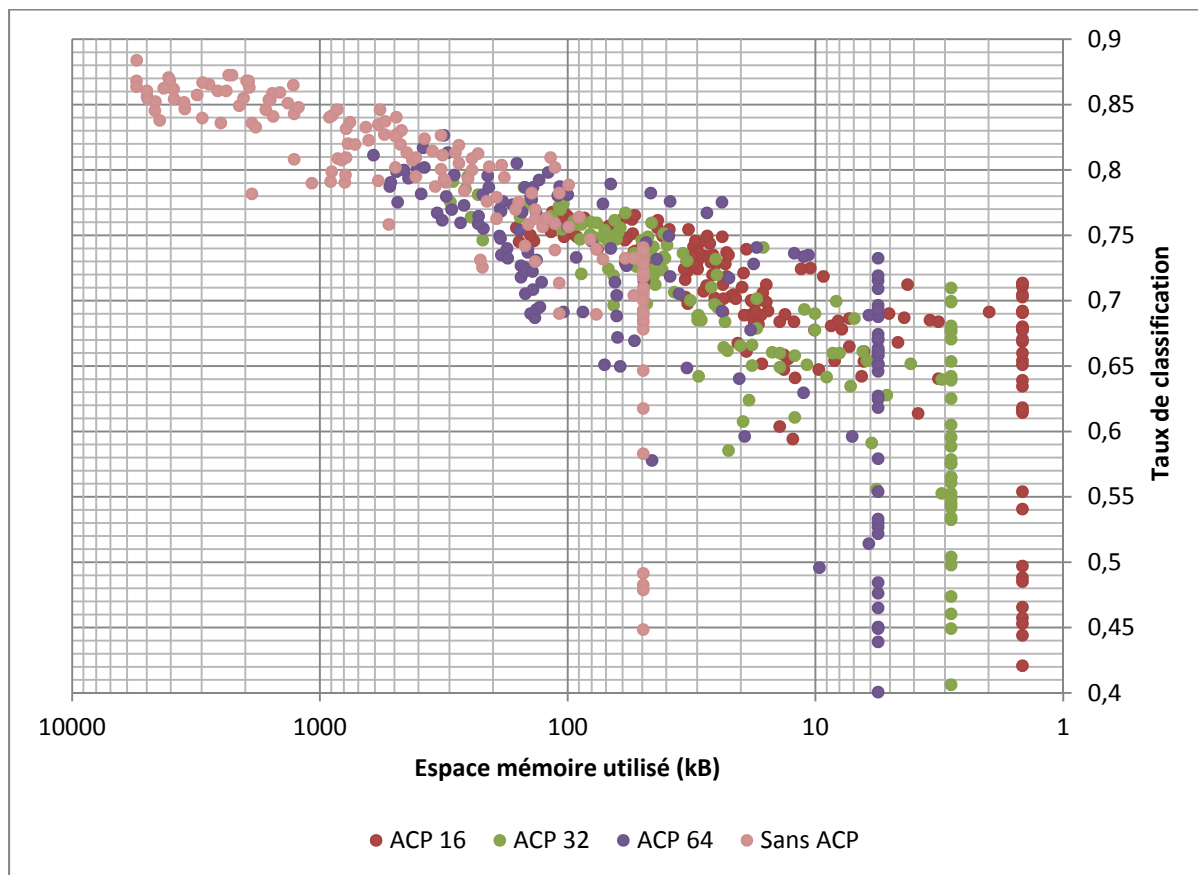


Figure 5.5 Fronts Pareto pour les différentes répliques statistiques en fonction du nombre de caractéristiques conservées dans le motif d'entrée (espace mémoire occupé en kB)

Le Tableau 5.1 et le Tableau 5.2 montrent respectivement le taux de classification moyen et l'espace moyen qu'occupe le réseau de neurones en mémoire ainsi que leur écart type associée d'un sous-groupe des solutions présentes dans les fronts de Pareto (Figure 5.4 &

Figure 5.5). Ces solutions types ont été choisies pour bien illustrer l'effet de la taille du réseau sur la performance de l'implémentation.

Tableau 5.1 Taux de classification moyen en généralisation sur la base de test

	Vocallo MGW		Atom N270		Core i3	
ACP 16 Légère	0.669	(0.031)	0.669	(0.031)	0.669	(0.031)
ACP 16 Moyenne	0.694	(0.027)	0.694	(0.027)	0.694	(0.027)
ACP 16 Lourde	0.747	(0.017)	0.748	(0.017)	0.748	(0.017)
ACP 32 Légère	0.621	(0.059)	0.621	(0.059)	0.621	(0.059)
ACP 32 Moyenne	0.650	(0.046)	0.650	(0.046)	0.650	(0.046)
ACP 32 Lourde	0.750	(0.019)	0.766	(0.019)	0.766	(0.019)
ACP 64 Légère	0.650	(0.070)	0.650	(0.070)	0.650	(0.070)
ACP 64 Moyenne	0.636	(0.070)	0.636	(0.070)	0.636	(0.070)
ACP 64 Lourde	0.742	(0.042)	0.783	(0.024)	0.783	(0.024)
Complète	--		0.844	(0.026)	0.844	(0.026)
Écart type entre parenthèses.						

Puisque le taux de classification et la compression sont déterminés durant la phase d'apprentissage, le choix du matériel a peu d'influence sur ceux-ci. L'exception est le processeur Vocallo MGW, qui imposait une contrainte de mémoire limitant la taille de la solution « volumineuse ». Ceci a donc limité la complexité du réseau utilisé et par conséquent diminué le taux de classification par rapport à l'implémentation pour l'Atom N270 et le Core i3-570.

En comparant les solutions produites en utilisant toute l'information des RI aux solutions de type « volumineuse » utilisant le maximum de caractéristiques conservées, une réduction de 781.5 % dans l'espace mémoire occupé par le réseau de neurones en ne subissant qu'une baisse de 6 % du taux de classification est observée.

Tableau 5.2 Espace mémoire moyen occupé
par le réseau entraîné

	Vocallo MGW		Atom N270		Core i3	
ACP 16 Légère	2.962	(2.0)	2.962	(2.0)	2.962	(2.0)
ACP 16 Moyenne	16.93	(4.6)	16.93	(4.6)	16.93	(4.6)
ACP 16 Lourde	66.91	(26.9)	88.37	(44.1)	88.37	(44.1)
ACP 32 Légère	3.635	(1.5)	3.635	(1.5)	3.635	(1.5)
ACP 32 Moyenne	14.67	(5.6)	14.67	(5.6)	14.67	(5.6)
ACP 32 Lourde	72.96	(33.0)	136.6	(65.4)	136.6	(65.4)
ACP 64 Légère	5.688	(0.04)	5.688	(0.04)	5.688	(0.04)
ACP 64 Moyenne	9.953	(6.0)	9.953	(6.0)	9.953	(6.0)
ACP 64 Lourde	66.12	(22.3)	344.7	(167.8)	344.7	(167.8)
Complète	--		2693.8	(1568)	2693.8	(1568)
Valeurs en kilobytes (kB). Écart type entre parenthèses.						

Le nombre de neurones présents dans le réseau a beaucoup plus d'influence sur le taux de classification et la taille du réseau que le nombre de caractéristiques conservées. Une plus grande amélioration du taux de classification est observée en comparant une solution « légère » à une solution « volumineuse » que lorsqu'une solution avec peu de caractéristiques est comparée avec une solution conservant un plus grand nombre. La même tendance se voit décrite pour l'espace mémoire bien que dans ce cas, l'effet du nombre de caractéristiques conservées est plus important.

Il est important de noter que l'espace mémoire occupé par un réseau de neurones augmente exponentiellement en fonction de sa complexité, mais que le taux de classification lui, augmente linéairement.

Le Tableau 5.3 et le Tableau 5.4 montrent respectivement le temps de traitement total pour une seule RI ainsi que la consommation énergétique pour le traitement de celle-ci par les trios processeurs. Contrairement au taux de classification et de la compression, le temps de traitement et la consommation énergétique sont intimement liés au choix du processeur utilisé.

Tableau 5.3 Temps d'exécution moyen
pour le traitement de la base de test

	Vocallo MGW		Atom N270		Core i3	
ACP 16 Légère	16.90	(11.2)	19.72	(13.1)	3.943	(2.38)
ACP 16 Moyenne	102.4	(27.9)	110.9	(30.4)	18.71	(4.41)
ACP 16 Lourde	554.2	(192.7)	574.4	(284.0)	94.53	(55.8)
ACP 32 Légère	21.1	(7.6)	23.06	(7.6)	4.432	(1.52)
ACP 32 Moyenne	84.37	(46.1)	96.66	(36.5)	17.00	(5.07)
ACP 32 Lourde	567.4	(192.7)	906.9	(405.7)	157.5	(71.0)
ACP 64 Légère	33.29	(1.21)	39.43	(2.39)	7.886	(1.93)
ACP 64 Moyenne	62.16	(39.0)	66.06	(37.0)	12.00	(5.58)
ACP 64 Lourde	656.2	(210.1)	2265	(1065)	388.5	(187.6)
Complète	--		17280	(9635)	2656	(1521)
Valeurs en microsecondes (μ s). Écart type entre parenthèses.						

Le deuxième facteur influençant le plus la consommation d'énergie et le temps total de traitement après le choix du matériel, est l'espace mémoire occupé par le réseau de neurones. Le fait que la consommation d'énergie et le temps total de traitement soient proportionnels à la taille du réseau renforce la pertinence d'utiliser la stratégie d'entraînement OMOEP pour les réseaux de neurones fuzzy ARTMAP. Cette technique permet de trouver le meilleur équilibre entre le taux de classification et la taille du réseau de neurones.

Il y a peu de différence entre le temps de traitement total obtenu à l'aide du processeur Vocallo MGW et celui obtenu à l'aide du processeur Atom N270. Par contre, le processeur Core i3-570 offre un temps d'exécution près de cinq fois plus rapide. Mais même en comparant la solution la plus « volumineuse » sur Vocallo MGW à la plus « légère » sur le Core i3-570, le processeur Vocallo MGW utilise neuf fois moins d'énergie. Lorsqu'on compare la solution « légère » appliquée sur les 2 mêmes processeurs, le Vocallo MGW consomme près de 200 fois moins d'énergie. En minimisant la taille du réseau de neurones et en choisissant adroitement le processeur pour exécuter l'algorithme, il est possible d'obtenir une réduction de la consommation d'énergie de plusieurs ordres de grandeur plus grande que la baisse du taux de classification.

**Tableau 5.4 Consommation d'énergie moyenne
pour le traitement de la base de test**

	Vocallo MGW		Atom N270		Core i3	
ACP 16 Légère	0.0052	(0.0036)	0.1217	(0.0850)	1.1232	(0.7849)
ACP 16 Moyenne	0.0294	(0.0082)	0.6886	(0.1922)	6.3578	(1.7751)
ACP 16 Lourde	0.1159	(0.0465)	3.5873	(1.7861)	33.122	(16.491)
ACP 32 Légère	0.0065	(0.0025)	0.1526	(0.0580)	1.4086	(0.5355)
ACP 32 Moyenne	0.0259	(0.0099)	0.6071	(0.2331)	5.6052	(2.1525)
ACP 32 Lourde	0.1283	(0.0576)	5.6312	(2.5590)	51.994	(23.628)
ACP 64 Légère	0.0103	(0.0006)	0.2416	(0.0144)	2.2309	(0.1330)
ACP 64 Moyenne	0.0179	(0.0107)	0.4187	(0.2506)	3.8659	(2.3138)
ACP 64 Lourde	0.1173	(0.0394)	14.315	(6.9473)	132.17	(64.146)
Complète	--		112.67	(65.601)	1040.3	(605.71)
Valeurs en joules (j). Écart type entre parenthèses.						

Il est intéressant de noter que tous les processeurs utilisés pour l'implémentation ont été capables de traiter les RI à une cadence plus rapide que la cadence d'acquisition des caméras NTSC, soit trente trames par seconde. En fait, le temps de traitement d'une solution sur le Vocallo MGW et l'Atom N270 peuvent avoir une cadence de soixante à deux mille fois plus grande lorsqu'ils utilisent les solutions « pesantes » ou « légères ». En utilisant le Core i3-530 et une solution de type « légère », la cadence de traitement peut être près de 6000 fois supérieure à la cadence d'acquisition de la caméra.

5.4 La performance en reconnaissance dans des séquences vidéo

La Figure 5.6 montre l'influence de la taille du tampon de décision sur le taux de classification moyen pour une séquence vidéo. Elle montre comment le taux de classification augmente avec la taille du tampon de décision pour atteindre 100 % lorsque le tampon est de taille suffisamment grande.

La séquence de test n° 2 est une exception et elle a été exclue de la moyenne pour le système. Lorsqu'on la compare aux autres classes d'identité, il y a très peu de données disponibles pour l'entraînement et l'évaluation de la performance. Après vérification, l'individu n° 2 de

la base de données est le seul individu venant d'un milieu ethnique différent. Ceci met à jour un problème fondamental du module de détection de visage.

Les premières implémentations du détecteur de visages de type Viola-Jones (Viola et Jones, 2004) ont été entraînées à l'aide d'images de visages recueillies lors d'une fouille sur internet. Dû à la prépondérance de la culture occidentale, la base de données d'entraînement a hérité d'un biais vers les visages de type caucasiens.

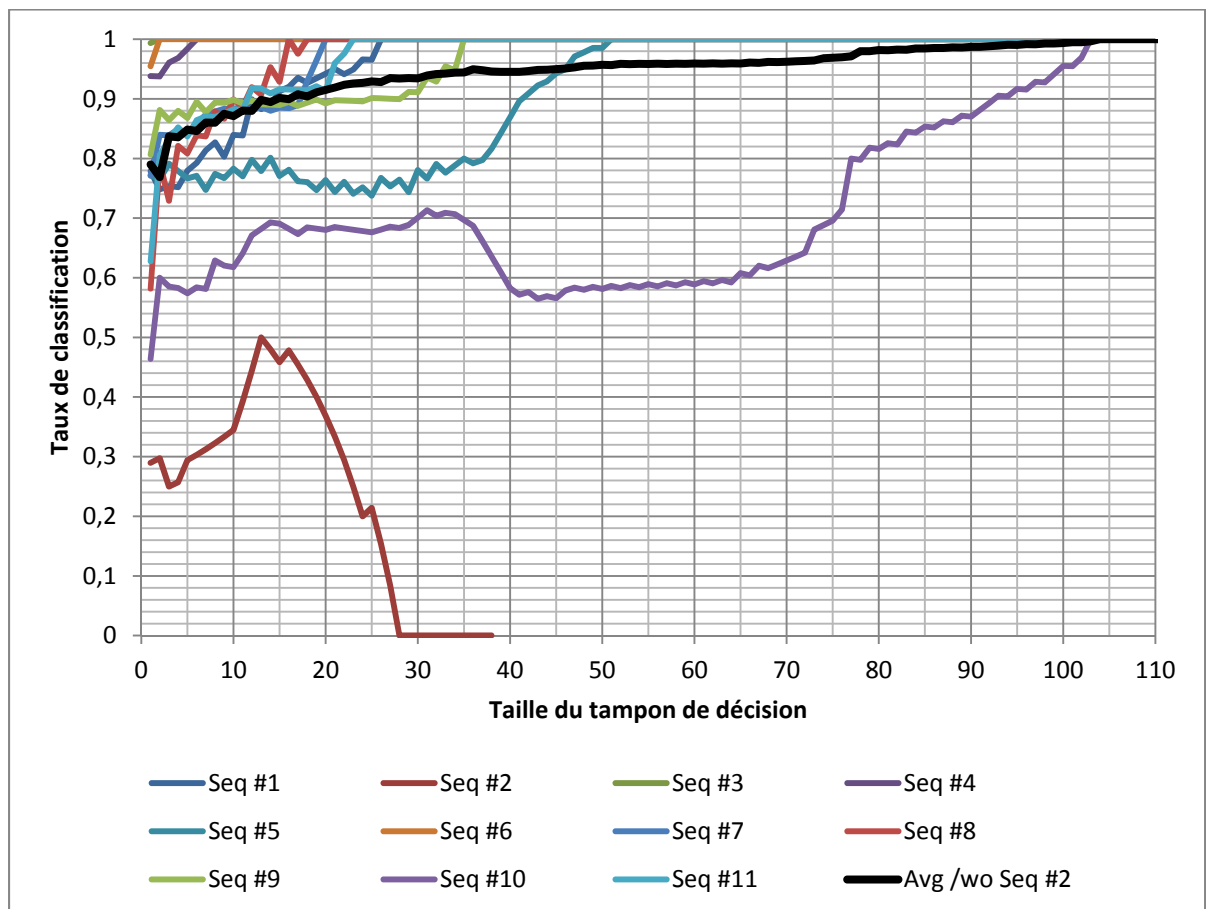


Figure 5.6 Taux de classification moyen en fonction de la taille du tampon de décision

En échange de l'amélioration dans le taux de classification, l'utilisation d'un tampon de décision crée un délai systématique avant de pouvoir produire une prédiction. Même dans le cas où le module de détection serait optimal, un individu doit rester dans le champ de vision

de la caméra pour plusieurs secondes pour avoir un taux de classification s'approchant de 100 % lorsqu'on utilise une cadence d'acquisition de 30 trames par secondes pour la caméra.

Les processeurs tels le Vocallo MGW et le Core i3 peuvent traiter les RI à une cadence plus élevée que celle de la caméra. Par conséquent, en utilisant une caméra ayant une cadence d'échantillonnage et un tampon de décision, il serait possible d'augmenter le taux de classification sans subir de grands délais dans la production de la prédiction finale.

5.5 Conclusion

Dans ce chapitre, il a été possible de voir que le temps utilisé par la communication intercœur est négligeable comparativement au temps nécessaire pour le calcul de la prédiction avec le Vocallo™ MGW et il y a peu de différences entre les différentes stratégies de communication évaluées. Il est aussi possible d'estimer le taux de classification d'un réseau de neurones fuzzy ARTMAP entraîné avec la stratégie OMOEP en fonction du nombre de caractéristiques conservées dans les motifs d'entrée en utilisant un classificateur de type k-PPV. L'espace mémoire occupé par le réseau de neurones fuzzy ARTMAP croît exponentiellement avec le nombre de caractéristiques conservées dans les motifs d'entrée.

L'influence du nombre de caractéristiques conservées dans les motifs d'entrée sur le taux de classification diminue lorsque la compression augmente. En utilisant l'espace mémoire nécessaire au lieu de la compression, il est possible de comparer des réseaux de neurones utilisant un différent nombre de caractéristiques dans les motifs d'entrée sur un même pied.

Il est possible d'obtenir une réduction de l'espace mémoire utilisé par le réseau de neurones d'un ordre de grandeur près de 100 fois plus grand que la baisse du taux de classification. Puisque le temps de traitement et la consommation énergétique sont proportionnels à l'espace mémoire utilisé par le réseau, ceci permet donc une forte réduction de la consommation d'énergie tout en favorisant un faible temps de traitement.

Il est aussi possible de rehausser le taux de classification en augmentant la taille du tampon de décision, mais ceci entraîne un délai systématique dans la production de la prédiction d'identité. Par contre, les implémentations multicœur ont un temps de traitement assez faible pour supporter une cadence d'acquisition plus élevée de la camera qui permettrait l'usage d'un tampon plus large, mais avec un plus faible délai dans la production de la prédiction de l'identité de l'individu présent dans la RI.

CONCLUSION

Comme il a été possible de voir dans cet ouvrage, la biométrie permet d'augmenter le niveau de sécurité en utilisant quelque chose d'intrinsèque à la personne au lieu de quelque chose qu'elle possède ou qu'elle connaît pour identifier un individu. Parmi les différents traits biométriques qu'un système d'identification automatique peut utiliser, la reconnaissance du visage se démarque par l'aisance d'acquisition et son acceptabilité sociale, mais elle n'est pas aussi discriminante que d'autres traits biométriques. L'analyse par composantes principales, ou ACP, permet de réduire la dimensionnalité des régions d'intérêt avant de les soumettre sous forme de motifs d'entrée tout en gardant le maximum d'informations pertinentes. Le classificateur neuronique fuzzy ARTMAP utilisé est un algorithme polyvalent qui offre un bon taux de classification et de compression tout en restant algorithmiquement simple dû à l'utilisation d'une distance L_1 et en remplaçant la multiplication par un opérateur fuzzy min basé sur des opérations de comparaison. Lorsque la technique d'apprentissage supervisé pour fuzzy ARTMAP avec OMOEP est utilisée, il est possible de produire un ensemble de réseaux de neurones entraînés qui représentent les meilleurs compromis entre le taux de classification et la compression.

Les trois processeurs typiques retenus pour représenter le spectre des solutions matérielles ont été le Core i3-530 d'Intel, un processeur trouvé dans les stations de travail, l'Atom-N270 aussi d'Intel, représentant les processeurs de miniportatifs, et le Vocallo MGW de la société Octasic, un processeur PSN de très basse consommation d'énergie. Les modifications apportées à l'algorithme fuzzy ARTMAP pour améliorer les performances lors de l'implémentation ont été le remplacement de la fonction de choix pour éliminer la division, l'encodage binaire de la couche F_{AB} , l'ajout d'un tampon de décision pour exploiter l'information spatio-temporelle ainsi que la modification pour permettre le traitement parallèle. Le taux de classification, l'espace mémoire utilisé par le réseau, le temps de traitement et la consommation énergétique des différentes implémentations ont été évalués à l'aide de la base de données vidéo IIT-CNRC.

Les résultats de la caractérisation de la communication intercœur sur le processeur Vocallo MGW ont montré que le temps passé à communiquer est négligeable par rapport au temps passé à calculer la prédiction. Ceci laisse donc une plus grande liberté de conception en laissant la possibilité de diviser un gros réseau en plusieurs sous-réseaux qui collaborent pour produire la prédiction ou bien pour utiliser plusieurs instances d'un plus petit réseau pour traiter plusieurs RI en parallèle.

L'expérience sur l'extraction et la sélection de caractéristiques a illustré l'existence d'un nombre optimal de caractéristiques à conserver, mais le fait que l'espace occupé en mémoire par le réseau croît exponentiellement avec le nombre de caractéristiques conservées force à choisir le nombre de caractéristiques minimales pour obtenir la performance désirée. Le taux de classification du réseau fuzzy ARTMAP une fois entraîné peut être estimé à l'aide du taux de classification obtenu à l'aide d'un classificateur de type k-PPV.

Pour bien voir les fronts de Pareto, la compression doit être présentée sur une échelle logarithmique et il est possible de comparer des réseaux de neurones utilisant différents nombres de caractéristiques en utilisant l'espace mémoire utilisé par le réseau au lieu de la compression. L'étude du taux de classification et de l'espace mémoire occupé par le réseau entraîné a montré que bien qu'il y a un prix à payer en exactitude de prédiction pour la réduction de dimensionnalité et en complexité du réseau, l'économie d'espace mémoire et d'énergie consommée est beaucoup plus grande. Les résultats montrent aussi que le nombre de neurones dans le réseau a plus d'influence que le nombre de caractéristiques conservées sur le taux de classification et qu'il est généralement plus efficace d'avoir plusieurs instances d'un petit réseau fonctionnant en parallèle qu'un seul réseau plus complexe. Finalement, l'utilisation d'un tampon de décision permet de rehausser le taux de classification au niveau désiré en échange d'un délai occasionné par l'accumulation des résultats de plusieurs RI d'une même séquence avant de produire la prédiction.

Si nous considérons l'identification continue, où le processus d'identification est répété à intervalles réguliers, au lieu d'une seule fois au début de l'utilisation, le délai causé par

l'utilisation d'un tampon de décision est moins dérangent que lorsque l'utilisateur doit attendre avant de pouvoir obtenir l'accès au système désiré. De façon similaire, la baisse de qualité de la prédiction est moins importante si le processus d'identification est utilisé plus pour des fins de personnalisation que de sécurité.

Il reste que l'utilisation de processeurs multicœurs de basse puissance permet d'avoir plusieurs instances d'un petit réseau de neurones traitant des RI en parallèle. Ceci permet d'obtenir une vitesse de traitement assez élevée pour accommoder une cadence d'acquisition plus rapide et ainsi réduire le délai nécessaire à la production de la prédiction à l'aide du tampon de décision, et ainsi combler la perte d'exactitude encourue par la réduction de la taille du réseau de neurones tout en minimisant la consommation d'énergie. Une plus grande vitesse de traitement permet aussi de traiter plusieurs RI présents dans la même trame simultanément.

Malgré toute l'attention et l'effort donné pour cette étude, plusieurs éléments d'intérêt n'ont pas pu être explorés en profondeur. Par exemple, le choix du nombre de caractéristiques conservées et le choix de la taille du tampon de décision auraient pu être intégrés dans le processus d'optimisation OMOEP. L'utilisation du logarithme de la compression ou la taille mémoire du réseau pourrait aussi aider le processus d'optimisation. Une base de données vidéo plus sophistiquée comportant plus d'individus et de séquences pour chacun d'eux permettrait d'avoir plus de variances intraclasse et éliminerait le besoin d'utiliser une validation croisée. Une telle base de données permettrait aussi de tester des scénarios où des individus non inscrits sont présentés au système pour identification. Le matériel disponible a aussi évolué durant le projet et il aurait été intéressant de pouvoir tester ces algorithmes sur des processeurs de type ARM qui ont beaucoup gagné en popularité dernièrement.

ANNEXE I

EXEMPLE DE CALCUL DE LA CONSOMMATION D'ÉNERGIE

Dans cette section, un exemple de calcul de la consommation d'énergie est effectué pour le processeur Intel Core i3-530 et la solution moyenne utilisant 16 composantes ACP. Les données propres aux problèmes sont présentées dans le Tableau-A I.1.

Tableau-A I.1 : Données pour le calcul de consommation d'énergie

Processeur	Intel Core i3-530
Nœuds de calculs (P)	2
Fréquence d'horloge (v)	2.53 GHz
Puissance (Ψ)	73 W
Type de solution	Moyenne
Nombre de composantes ACP	16
Nombre moyen de neurones	127.5
Nombre de motifs	1585

Le nombre d'opérations (χ) trouvées dans le code assembleur et exécutées par le processeur pour traiter la base de données est donné par l'équation I.1 :

$$\chi = t(34d + f(132d + 64) + 53) \quad (I.1)$$

La consommation d'énergie (κ) par le processeur pour le traitement de la base de données est donnée par l'équation I.2 :

$$\kappa = \chi\Psi/(Pv) \quad (I.2)$$

Ainsi, l'énergie consommée par le processeur Intel Core i3-530 pour le traitement de la base de test en utilisant la solution moyenne avec 16 composantes ACP est :

$$\begin{aligned}
\kappa &= \chi \Psi / (Pv) \\
\kappa &= t (34d + f(132d + 64) + 53) \Psi / (Pv) \\
\kappa &= (1585) (34(16) + (127.5) (132(16) + 64) + 53) (73W) / ((2)(2.53GHz)) \\
\kappa &= 6.358 \text{ Joules}
\end{aligned} \tag{I.3}$$

LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES

- Abate, A. F., M. Nappi, D. Riccio et G. Sabatino. 2007. « 2D and 3D face recognition: A survey ». *Pattern Recogn. Lett.*, vol. 28, n° 14, p. 1885-1906.
- Aggarwal, G., A. Veeraraghavan et R. Chellappa. 2005. « 3D facial pose tracking in uncalibrated videos ». In *Proceedings of the First international conference on Pattern Recognition and Machine Intelligence*. (Kolkata, India), p. 515-520. Springer-Verlag.
- Ahonen, T., A. Hadid et M. Pietikainen. 2006. « Face Description with Local Binary Patterns: Application to Face Recognition ». *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, n° 12, p. 2037-2041.
- Arandjelovic, O., O. Ar et R. Cipolla. 2006. « Face Recognition from Video Using the Generic Shape-Illumination Manifold ». In *9th European Conference on Computer Vision*. (Graz, Austria, May 7-13, 2006) Vol. 3954, p. 27-40. Springer.
- Baek, K., B. Draper, R. Beveridge et K. She. 2002. « PCA vs. ICA: A Comparison on the FERET Data Set ». In *4th International Conference on Computer Vision, Pattern Recognition and Image Processing in conjunction with the 6th JCIS*. (Durham, NC, 8-14 March, 2002), p. 824-827.
- Bardeen, J., et W. H. Brattain. 1948. « The Transistor, A Semi-Conductor Triode ». *Physical Review*, vol. 74, n° 2, p. 230-231.
- Bartlett, M. S., J. R. Movellan et T. J. Sejnowski. 2002. « Face recognition by independent component analysis ». *Neural Networks, IEEE Transactions on*, vol. 13, n° 6, p. 1450-1464.
- Batur, A. U., B. E. Flinchbaugh et M. H. Hayes, III. 2003. « A DSP-based approach for the implementation of face recognition algorithms ». In *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*. (6-10 April 2003) Vol. 2, p. 253-6.
- Belhumeur, P. N., J. P. Hespanha et D. J. Kriegman. 1997. « Eigenfaces vs. Fisherfaces: recognition using class specific linear projection ». *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, n° 7, p. 711-720.
- Blume, M., et S. C. Esener. 1997. « An Efficient Mapping of Fuzzy ART onto a Neural Architecture ». *Neural Networks*, vol. 10, n° 3, p. 409-411.

- Borgatti, M., F. Lertora, B. Foret et L. Cali. 2003. « A reconfigurable system featuring dynamically extensible embedded microprocessor, FPGA, and customizable I/O ». *Solid-State Circuits, IEEE Journal of*, vol. 38, n° 3, p. 521-529.
- Canturk, I., B. Alper, C. Chen-Yong, B. Pradip et M. Margaret. 2006. « An Analysis of Efficient Multi-Core Global Power Management Policies: Maximizing Performance for a Given Power Budget ». In *Microarchitecture, 2006. MICRO-39. 39th Annual IEEE/ACM International Symposium on*. (Dec. 2006), p. 347-358.
- Carpenter, G. A., et M. N. Gjaja. 1993. *Fuzzy ART choice functions*. Coll. « Technical report CAS/CNS ; », 93-060. Boston, MA: Boston University, Center for Adaptive Systems and Dept. of Cognitive and Neural Systems, 14 p.
- Carpenter, G. A., S. Grossberg, N. Markuzon, J. H. Reynolds et D. B. Rosen. 1992. « Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps ». *Neural Networks, IEEE Transactions on*, vol. 3, n° 5, p. 698-713.
- Carpenter, G. A., B. L. Milenova et B. W. Noeske. 1998. « Distributed ARTMAP: a neural network for fast distributed supervised learning ». *Neural Networks*, vol. 11, n° 5, p. 793-813.
- Carpenter, G. A., et W. D. Ross. 1995. « ART-EMAP: A neural network architecture for object recognition by evidence accumulation ». *Neural Networks, IEEE Transactions on*, vol. 6, n° 4, p. 805-18.
- Castro, J., J. Secretan, M. Georgiopoulos, R. DeMara, G. Anagnostopoulos et A. Gonzalez. 2007. « Pipelining of Fuzzy ARTMAP without matchtracking: Correctness, performance bound, and Beowulf evaluation ». *Neural Networks*, vol. 20, n° 1, p. 109-128.
- Castro, J., M. Georgiopoulos, J. Secretan, R. F. Demara, G. Anagnostopoulos et A. J. Gonzalez. 2004. « Parallelization of Fuzzy ARTMAP to Improve its Convergence Speed: The Network Partitioning Approach and the Data Partitioning Approach ». *Nonlinear Analysis*, vol. 63, p. 877-889.
- Chan, A. B., et N. Vasconcelos. 2005. « Probabilistic kernels for the classification of autoregressive visual processes ». In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. (20-25 June 2005) Vol. 1, p. 846-851.
- Chellappa, R., M. Bicego et P. Turaga. 2009. « A Review of Video-Based Face Recognition Algorithms ». In *Handbook of Remote Biometrics*, sous la dir. de Tistarelli, Massimo, Stan Z. Li et Rama Chellappa. p. 193-216. Coll. « Advances in Pattern Recognition ». London: Springer

- Coello, C. A. C., G. T. Pulido et M. S. Lechuga. 2004. « Handling multiple objectives with particle swarm optimization ». *Evolutionary Computation, IEEE Transactions on*, vol. 8, n° 3, p. 256-279.
- Constantinou, T., Y. Sazeides, P. Michaud, D. Fetis et A. Sez nec. 2005. « Performance implications of single thread migration on a chip multi-core ». *SIGARCH Comput. Archit. News*, vol. 33, n° 4, p. 80-91.
- de Luis-García, R., C. Alberola-López, O. Aghzout et J. Ruiz-Alzola. 2003. « Biometric identification systems ». *Signal Processing*, vol. 83, n° 12, p. 2539-2557.
- Duda, R. O., P. E. Hart et D. G. Stork. 2001. *Pattern classification*, 2nd Edition. Wiley-Interscience, 654 p.
- Golfarelli, M., D. Maio et D. Malton. 1997. « On the error-reject trade-off in biometric verification systems ». *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, n° 7, p. 786-796.
- Gorodnichy, D. O. 2006. « Editorial: Seeing faces in video by computers. Editorial for Special Issue on Face Processing in Video Sequences ». *Image Vision Comput.*, vol. 24, n° 6, p. 551-556.
- Granger, É., P. Henniges, R. Sabourin et L. S. Oliveira. 2007. « Supervised Learning of Fuzzy ARTMAP Neural Networks Through Particle Swarm Optimisation ». *Journal of Pattern Recognition Research*, vol. 2, n° 1, p. 27-60.
- Granger, É., D. Prieur et J.-F. Connolly. 2010. « Evolving ARTMAP neural networks using Multi-Objective Particle Swarm Optimization ». In *Evolutionary Computation (CEC), 2010 IEEE Congress on*. (18-23 July 2010), p. 1-8.
- Hadid, A., et M. Pietikainen. 2005. « An Experimental Investigation about the Integration of Facial Dynamics in Video-Based Face Recognition ». *Electronic Letters on Computer Vision and Image Analysis; Vol 5, No 1: August 2005*.
- Intel. 2008. « First the Tick, Now the Tock: Next Generation Intel® Microarchitecture (Nehalem) ». En ligne. 8 p. < <http://www.intel.com/technology/architecture-silicon/next-gen/whitepaper.pdf> >. Consulté le 17 mars 2011.
- Intel. 2009a. « Intel Atom N270 Processor Specification ». En ligne. 57 p. < <http://www.intel.com/content/dam/doc/datasheet/mobile-atom-n270-single-core-datasheet-.pdf> >. Consulté le 11 novembre 2010.

- Intel. 2009b. « Intel Core i3-530 Processor Specification ». En ligne. 2 p. < [<http://ark.intel.com/products/46472/Intel-Core-i3-530-Processor-\(4M-Cache-2_93-GHz\)>](http://ark.intel.com/products/46472/Intel-Core-i3-530-Processor-(4M-Cache-2_93-GHz)) >. Consulté le 19 novembre 2010.
- Jain, A. K., A. Ross et S. Pankanti. 2006. « Biometrics: a tool for information security ». *Information Forensics and Security, IEEE Transactions on*, vol. 1, n° 2, p. 125-143.
- Jain, A. K., A. Ross et S. Prabhakar. 2004. « An introduction to biometric recognition ». *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 14, n° 1, p. 4-20.
- Jebara, T. S., et A. Pentland. 1997. « Parametrized structure from motion for 3D adaptive feedback tracking of faces ». In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*. (17-19 Jun 1997), p. 144-150.
- Kennedy, J., et R. Eberhart. 1995. « Particle swarm optimization ». In *Neural Networks, 1995. Proceedings., IEEE International Conference on*. (Nov/Dec 1995) Vol. 4, p. 1942-1948 vol.4.
- Kilby, J. S. 1976. « Invention of the integrated circuit ». *Electron Devices, IEEE Transactions on*, vol. 23, n° 7, p. 648-654.
- Kleihorst, R., A. Abbo, B. Schueler et A. Danilin. 2007. « Camera mote with a high-performance parallel processor for real-time frame-based video processing ». In *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*. (5-7 Sept. 2007), p. 69-74.
- Kondo, H., M. Nakajima, M. Z. Bober, K. Kucharski, O. Yamamoto et T. Shimizu. 2005. « Implementation of Face Recognition Processing using an Embedded Processor ». *J. Robotics & Mechatronics*, vol. 17, n° 4.
- Kung, S.Y., W. Mak et S.H. Lin. 2005. *Biometric authentication: a machine learning approach*. Prentice Hall Professional Technical Reference, 476 p.
- Lee, K.-C., J. Ho, M.-H. Yang et D. J. Kriegman. 2005. « Visual tracking and recognition using probabilistic appearance manifolds ». *Computer Vision and Image Understanding*, vol. 99, n° 3, p. 303-331.
- Li, B., et R. Chellappa. 2001. « Face verification through tracking facial features ». *J. Opt. Soc. Am. A*, vol. 18, n° 12, p. 2969-2981.

- Lubkin, J., et G. Cauwenberghs. 1999. « VLSI Implementation of Fuzzy Adaptive Resonance and Learning Vector Quantization ». In *Proceedings of the 7th International Conference on Microelectronics for Neural, Fuzzy and Bio-Inspired Systems*. p. 147. IEEE Computer Society.
- Malkani, A., et C. A. Vassiliadis. 1995. « Parallel implementation of the fuzzy ARTMAP neural network paradigm on a hypercube ». *Expert Systems*, vol. 12, n° 1, p. 39-53.
- Mao, W., et A. Bigdeli. 2004. « Implementation of a real-time automated face recognition system for portable devices ». In *Communications and Information Technology, 2004. ISCIT 2004. IEEE International Symposium on*. (26-29 Oct. 2004) Vol. 1, p. 89-92.
- Martinez, A. M., et A. C. Kak. 2001. « PCA versus LDA ». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, p. 228-233.
- Matta, F., et J.-L. Dugelay. 2009. « Person recognition using facial video information: A state of the art ». *J. Vis. Lang. Comput.*, vol. 20, n° 3, p. 180-187.
- Nagel, J.-L., P. Stadelmann, M. Ansorge et F. Pellandini. 2002. « A Low-Power VLSI Architecture for Face Verification Using Elastic Graph Matching ». In *11th European Signal Processing Conference*. (Toulouse, France, 03-06 septembre 2002) Vol. III, p. 577-580.
- Niinuma, K., P. Unsang et A. K. Jain. 2010. « Soft Biometric Traits for Continuous User Authentication ». *Information Forensics and Security, IEEE Transactions on*, vol. 5, n° 4, p. 771-780.
- Octasic. 2004. « Vocallo MGW: The Expendable Media Gateway Solution ». En ligne. 8 p. <
<http://www.octasic.com/documents/en/products/vocallo/octvocpb2004.pdf>
 >. Consulté le 4 décembre 2008.
- Prabhakar, S., S. Pankanti et A. K. Jain. 2003. « Biometric recognition: security and privacy concerns ». *Security & Privacy, IEEE*, vol. 1, n° 2, p. 33-42.
- Reyneri, L. M. 2003. « Implementation issues of neuro-fuzzy hardware: going toward HW/SW codesign ». *Neural Networks, IEEE Transactions on*, vol. 14, n° 1, p. 176-194.
- Ross, A. A., K. Nandakumar et A. K. Jain. 2006. *Handbook of multibiometrics*, 1st Edition. Coll. « International Series on Biometrics ». New York, NY: Springer, 224 p.
- Sirovich, L., et M. Kirby. 1987. « Low-Dimensional Procedure for the Characterization of Human Faces ». *Journal of the Optical Society of America A*, vol. 4, n° 3, p. 519-524.

- Stallkamp, J., H. K. Ekenel et R. Stiefelhagen. 2007. « Video-based Face Recognition on Real-World Data ». In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. (14-21 Oct. 2007), p. 1-8.
- Tistarelli, M., M. Bicego et E. Grosso. 2009. « Dynamic face recognition: From human to machine vision ». *Image and Vision Computing*, vol. 27, n° 3, p. 222-232.
- Tremblay, J. P. 2009. « Analyse de performance multi-niveau et partitionnement d'application radio sur une plateforme multiprocesseur ». Mémoire de maîtrise en génie électrique. Montréal, École polytechnique de Montréal, 117 p.
- Turk, M. A., et A. P. Pentland. 1991. « Face recognition using eigenfaces ». In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*. (3-6 Jun 1991), p. 586-591.
- Viola, P., et M. J. Jones. 2004. « Robust Real-Time Face Detection ». *Int. J. Comput. Vision*, vol. 57, n° 2, p. 137-154.
- Wayman, J. L. . 2001. « Fundamentals of Biometric Authentication Technologies ». *International Journal of Image and Graphics*, vol. 1, n° 1, p. 93-113.
- Wiskott, L., J. M. Fellous, N. Kuiger et C. von der Malsburg. 1997. « Face recognition by elastic bunch graph matching ». *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, n° 7, p. 775-779.
- Xiaoming, L., et C. Tsuhan. 2003. « Video-based face recognition using adaptive hidden Markov models ». In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*. (18-20 June 2003) Vol. 1, p. I-340-I-345 vol.1.
- Yao, X. 1999. « Evolving Artificial Neural Networks ». *PIEEE: Proceedings of the IEEE*, vol. 87, p. 1423-1447.
- Zhao, W. , R. Chellappa, P. J. Phillips et A. Rosenfeld. 2003. « Face recognition: A literature survey ». *ACM Comput. Surv.*, vol. 35, n° 4, p. 399-458.
- Zhou, A., B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan et Q. Zhang. 2011. « Multiobjective evolutionary algorithms: A survey of the state of the art ». *Swarm and Evolutionary Computation*, vol. 1, n° 1, p. 32-49.
- Zhou, S. K., V. Krueger et R. Chellappa. 2003. « Probabilistic recognition of human faces from video ». *Computer Vision and Image Understanding*, vol. 91, n° 1-2, p. 214-245.

